

Universidad de Alcalá
Escuela Politécnica Superior

**GRADO EN INGENIERÍA EN ELECTRÓNICA DE
COMUNICACIONES**



Trabajo Fin de Grado

Sistema de control remoto en quesería artesana

**ESCUELA POLITECNICA
SUPERIOR**

Autor: JORGE TRIVIÑO ROMANO

Tutor/es: MARTA MARRÓN ROMERA

2018

UNIVERSIDAD DE ALCALÁ
ESCUELA POLITÉCNICA SUPERIOR

**GRADO EN INGENIERÍA ELECTRÓNICA DE
COMUNICACIONES**

TRABAJO FIN DE GRADO

AUTOR: JORGE TRIVIÑO ROMANO

TUTORA: MARTA MARRÓN ROMERA

TRIBUNAL:

PRESIDENTE: JULIO PASTOR MENDOZA

VOCAL 1º: PEDRO ALFONSO REVENGA TORO

VOCAL 2º: MARTA MARRÓN ROMERA

FECHA:

Índice

Índice	4
Índice de figuras	8
Resumen.....	12
Abstract	14
1 Introducción	16
1.1 Alcance	16
1.2 Contenido	16
2 Análisis.....	18
2.1 Descripción del problema	18
2.2 Listado de requisitos y especificaciones.	18
2.3 Descripción del sistema.....	19
2.4 Análisis del sistema	20
2.4.1 Descripción de la cámara de curación.....	21
2.4.2 Adquisición de datos. Sensores y dispositivos	22
2.4.3 Comunicaciones internas	27
2.4.4 Sistema de control e interfaz de usuario	30
2.4.5 Actuadores	33
2.4.6 Comunicaciones externas	34
3 Diseño.....	36
3.1 Red de comunicación interna.....	36
3.1.1 Zigate	37
3.1.2 Comunicación controlador – sensores.....	37
3.1.3 Comunicación controlador – actuador.....	38
3.1.4 Comunicación controlador - <i>router</i>	39
3.2 Red de comunicación externa.....	39
3.3 Sensor de temperatura y humedad	40
3.3.1 Datos meteorológicos externos	40
3.3.2 Ubicación de los sensores	40
3.4 Actuadores	41
3.5 Hardware.....	42
3.6 Software de control.....	43
3.7 Automatización	43
3.8 Interfaz de usuario	44
4 Implementación en la quesería artesana.....	46
4.1 Configuración de “Home Assistant”	46
4.1.1 Instalación del sistema operativo.....	46

4.1.2	Instalación de “Home Assistant”	49
4.1.3	Configuración del sensor de temperatura y humedad	49
4.1.4	Configuración del servidor de datos de climatología exterior	51
4.1.5	Configuración del actuador radiador de calor	52
4.1.6	Configuración del actuador del motor de frío.....	53
4.1.7	Configuración del acceso externo	55
4.1.8	Configuración del auto-inicio de HA.....	57
4.2	Notificaciones de “Home Assistant”	57
4.3	Automatización de “Home Assistant”	57
4.3.1	Notificación de reinicio de “Home Assistant”	58
4.3.2	Notificación de anomalías.....	58
4.3.3	Control de temperatura	60
4.3.4	Control de humedad	60
4.3.5	Ahorro de energía	61
4.4	Historial de “Home Assistant”	62
5	Pruebas.....	64
5.1	Temperatura alta en la cámara de curación	64
5.2	Correcta climatización.....	65
5.3	Correcto funcionamiento del equipo de frío	65
6	Conclusiones y líneas de trabajo abiertas.....	66
6.1	Conclusiones.....	66
6.2	Líneas futuras	66
7	Presupuesto	68
7.1	Material que compone el sistema.....	68
7.2	Recursos hardware.....	68
7.3	Recursos software	68
7.4	Coste de la mano de obra	69
7.5	Presupuesto de ejecución material.....	69
7.6	Importe de la ejecución por contrata	69
7.7	Honorarios facultativos	69
7.8	Presupuesto total	70
	Referencias.....	72
	Anexo A. Manual de usuario	74
	A.1. Acceso a la interfaz	74
	A.1.1. Desde dentro de la red local	74
	A.1.2. Desde el exterior de la red local	74
	A.2. Contraseña de acceso	74
	A.3. Pantalla de resumen	74

A.4. Pantalla de historial	75
----------------------------------	----

Índice de figuras

Figura 2-1 - Diagrama del sistema.....	20
Figura 2-2 - Diagrama básico del sistema.....	20
Figura 2-3 - Cámara de curación de quesos	21
Figura 2-4 - PID IR33.....	22
Figura 2-5 - Sensor termo-higrómetro CAREL.....	22
Figura 2-6 - Diseño 3D de las instalaciones.....	23
Figura 2-7 - Sensor DS18B20	23
Figura 2-8 – Sensor LM35.....	24
Figura 2-9 - Sensor DHT22.....	24
Figura 2-10 - Sensor SHT30	25
Figura 2-12 - Sensor “XIAOMI”	26
Figura 2-11 - Sensor “NEST”	26
Figura 2-13 - Sensor “SONOFF”	26
Figura 2-14 - Tabla de características GSM/2G/3G/4G.....	27
Figura 2-15 - Principales estándares WiFi	28
Figura 2-16 - Características de WiFi Halow.....	28
Figura 2-17 - Características de Bluetooth.....	29
Figura 2-18 - Características de Z-Wave.....	29
Figura 2-19 - Datos de búsquedas en Google	30
Figura 2-20 - Características de ZigBee	30
Figura 2-21 - Interfaz gráfica de OpenHAB.....	31
Figura 2-22 - Interfaz gráfica de “HomeAssistant”, HA.....	32
Figura 2-23 – Imagen de la “Raspberry pi 3”	33
Figura 2-24 - Actuador “OSRAM SMART PLUG”.....	34
Figura 2-25 - Actuador “SONOFF” de “ITEAD”	34
Figura 2-26 - Actuador “TP-LINK”	34
Figura 3-1 - Diagrama de los dispositivos inalámbricos	36
Figura 3-2 - Interior del Zigate.....	37
Figura 3-3 - Diagrama de comunicación controlador – sensores.....	38
Figura 3-4 - Diagrama de comunicación controlador-actuador	38
Figura 3-5 - Capas del protocolo MQTT	39
Figura 3-6 - Diagrama de bloques del SAD y vista física interior.....	41
Figura 3-7 - Dispositivo final SAD	41
Figura 3-8 - Diagrama de actuadores del sistema.....	42

Figura 3-9 – Diagrama de conexiones del hardware de control	43
Figura 3-10 - Pestaña principal de “Home Assistant”	45
Figura 3-11 - Pestaña del historial de “Home Assistant”	45
Figura 4-1 - Diagrama global del sistema completo.....	46
Figura 4-2 - Imagen del programa “Etcher”	47
Figura 4-3- Interfaz WINSOCP	47
Figura 4-4- Captura del interfaz de Putty.....	48
Figura 4-5- Captura de trabajo con terminal con Putty	48
Figura 4-6- Archivos del componente Zigate	50
Figura 4-7-Configuración de sensores Zigate.....	50
Figura 4-8- Ventana de servicios de HA	50
Figura 4-9- Valor de la dirección del dispositivo ZigBee	51
Figura 4-10- Visualización del sensor de temperatura y humedad	51
Figura 4-11- Archivo de configuración de “darksky”	52
Figura 4-12- Visualización de datos climatológicos.....	52
Figura 4-13- Configuración del interruptor “actuador 1”	53
Figura 4-14- Visualización del interruptor “resistencia de calor”	53
Figura 4-15- Configuración del <i>broker</i> “Mosquitto”	54
Figura 4-16- Menú principal de “Tasmota”	55
Figura 4-17- Configuración de MQTT en “Tasmota”	55
Figura 4-18- Configuración del actuador en HA.....	56
Figura 4-19- Visualización de los interruptores de control de los actuadores.....	56
Figura 4-20- Interfaz de configuración de los puertos del <i>router</i>	56
Figura 4-21- Configuración de notificaciones vía “Telegram”	57
Figura 4-22 - Envío de notificación de prueba en HA.....	58
Figura 4-23- Notificación de “Telegram” recibida.....	58
Figura 4-24- Configuración de reinicio de HA	59
Figura 4-25- Configuración de notificación de temperatura alta.....	59
Figura 4-26 – Configuración de notificación de temperatura baja	59
Figura 4-27- Configuración de notificación de humedad alta.....	59
Figura 4-28- Configuración de notificación de humedad baja	60
Figura 4-29- Configuración del componente <i>generic_thermostat</i>	60
Figura 4-30- Configuración del control de humedad en HA	61
Figura 4-31- Captura de la automatización para el ahorro energético.....	61
Figura 4-32- Configuración del historial.....	62
Figura 4-33- Historial gráfico de las medidas	62

Figura 5-1- Gráfica de temperatura	64
Figura 5-2 – Notificación de “Telegram” de temperatura superior a 15°C	64
Figura 5-3 - Temperatura en el interior de la cámara de curación	65
Figura 5-4- Gráfica salida equipo de frío	65
Figura A-1- Captura introducir contraseña	74
Figura A-2- Pantalla resumen	75
Figura A-3- Captura del historial	75

Resumen

Empleando software libre se ha diseñado un sistema de control remoto capaz de gestionar en tiempo real la cámara de curación de una quesería artesana. Los datos se muestran de manera gráfica a través de una interfaz de usuario, con el fin de facilitar su análisis ante posibles anomalías en la instalación.

El diseño del sistema de control incluye diferentes automatizaciones con el fin de mejorar la climatización, notificar anomalías en tiempo real o ahorrar energía.

Palabras Clave: IoT, Zigbee, MQTT, HomeAssistant, Raspberry, Tasmota, Zigate, software libre, queso artesano, control remoto

Abstract

A remote control system for real time management of the curing chamber of an artisan cheese factory has been designed in this thesis. Free software has been used for this purpose. The obtained data are shown in a graphic user interface, enabling its analysis to determine possible malfunctions in the installation.

Automations have been included in the control system to improve the air conditioning of the installation, notify anomalies in real time and enhance the energy efficiency of the factory.

1 Introducción

El objetivo principal de este documento es servir como memoria del proceso de desarrollo del trabajo fin de grado: *“Sistema de control remoto en quesería artesana”*.

La elaboración del documento se ha realizado empleando un enfoque profesional con la finalidad de proponer una solución documentada a un posible cliente. También destaca la abundante información relativa a la implantación del proyecto, siendo este último uno de sus objetivos.

En el sector de la alimentación, tanto la temperatura como la humedad siempre han sido dos de los factores más importantes a tener en cuenta. Esto es debido a la importancia de elaborar un producto de características regulares a lo largo del año. En la fase de maduración del queso es, si cabe, aún más importante, ya que se depende directamente del trabajo que hacen las bacterias lácticas en todo momento.

Ante esta necesidad, se pretende controlar la temperatura y humedad de una cámara de curación de una manera energéticamente eficiente, siendo capaz de examinar y actuar en tiempo real ante las adversidades que puedan surgir durante el proceso de maduración. En resumen, el objetivo final del proyecto es diseñar e implementar un sistema de control capaz de actuar sobre el funcionamiento del motor de frío de una cámara de curación. La actuación se ejecuta de forma autónoma en función de diferentes datos de entrada, tales como sensores de temperatura y humedad ubicados en el interior de la instalación, así como datos meteorológicos externos.

Con el fin de facilitar su implantación se pretende utilizar un sistema de comunicación inalámbrico, reduciendo al mínimo los costes de obra. Además, todo el sistema se controla mediante una interfaz de usuario desde cualquier punto con conexión a internet.

El sistema final es plenamente exportable a cualquier otro proceso de control o monitorización del panorama IoT.

1.1 Alcance

El control remoto de instalaciones forma parte del mundo IoT (*“Internet of Things”*). Se trata de un tema muy amplio en el que están implicadas varias disciplinas técnicas y científicas y con una gran repercusión social, lo que propicia un avance extremadamente rápido de la tecnología en esta área.

El documento está desarrollado con el fin de proporcionar una solución profesional al problema planteado por un posible cliente. Por lo tanto, se limitará la extensión a la información necesaria para cubrir este objetivo añadiendo las referencias a información complementaria cuando sea necesario.

1.2 Contenido

Este documento se encuentra estructurado de manera que el lector pueda entender e incluso implementar la solución desarrollada al problema planteado por el cliente. A continuación, se explican brevemente cada una de estas etapas.

a) Fase de Análisis – Capítulo 2

En este capítulo se incluye la descripción de la problemática a resolver. Así como la documentación de las posibles opciones que plantea el panorama tecnológico actual para el desarrollo del proyecto que ofrezca una solución a dicha problemática.

b) Fase de Diseño – Capítulo 3

Esta fase se centra en el proceso de elaboración de una solución al problema planteado en función de las especificaciones de requisitos elaboradas anteriormente. En este apartado se exponen las decisiones tomadas para la construcción del sistema y se explican los motivos por los que se han tomado. El capítulo se encuentra dividido en diferentes secciones que corresponden con las distintas partes del sistema final.

c) Fase de Implementación – Capítulo 4

Se explica el proceso llevado a cabo para la construcción e implementación del sistema. Se incluyen los detalles necesarios para la configuración del sistema, como por ejemplo la manera de instalar y programar el software de control.

d) Fase de Pruebas – Capítulo 5

Se explica el modo en que el sistema se comporta durante la ejecución y cómo reacciona ante determinadas pruebas que se han llevado a cabo para comprobar su correcto funcionamiento y el cumplimiento de los requisitos.

e) Conclusiones y Líneas Futuras – Capítulo 6

En este apartado se analiza el cumplimiento de todos los objetivos formulados al inicio del proyecto. En último lugar, se describen brevemente las futuras líneas de trabajo que surgen de este proyecto.

2 Análisis

2.1 Descripción del problema

El control remoto de instalaciones en el mundo de la alimentación es todavía poco usual, lo cual no quiere decir que no sea importante, todo lo contrario. La detección precoz de fallos o anomalías en las instalaciones puede suponer mucho ahorro de dinero, ya que en la mayoría de casos los productos son perecederos o simplemente evolucionan de una manera diferente en función de las condiciones ambientales.

El sistema propuesto en esta memoria se centra en el control remoto de dos variables, temperatura y humedad, en la cámara de curación de una quesería artesana. Ambas variables son fundamentales en el proceso de curación de los quesos.

- **¿Cómo afecta la temperatura en la curación de los quesos duros?**

Para el tipo de queso elaborado en la quesería artesana estudiada en este proyecto, la temperatura ideal de secado tiene que estar entre los 12°C y 14°C. Una temperatura menor hará que el proceso de curación sea más lento, afectando también a la correcta proteólisis e hidrólisis (degradación de las grasas). Una temperatura mayor a los 15°C puede provocar un secado excesivo, perjudicando también a los procesos químicos mencionados anteriormente.

- **¿Cómo afecta la humedad en la fase de curación de los quesos duros?**

La humedad relativa es el otro factor más importante a tener en cuenta en el proceso de maduración de los quesos. Para el tipo de queso bajo estudio en este proyecto, la humedad debe estar entre un 80 y 85%. Una humedad superior provocará problemas en el desuerado y un excesivo crecimiento de mohos en la corteza. En el caso contrario, si la humedad baja del 80% se producirá un secado excesivo, teniendo como causa principal la merma en el peso del producto final.

Hasta ahora la instalación no cuenta con ningún dispositivo capaz de obtener un histórico de temperaturas y humedad, siendo asistir a la quesería la única opción de comprobar anomalías. Ese periodo de tiempo, que en el caso de la quesería de estudio puede llegar a ser de 3 o 4 días en alguna ocasión, puede provocar graves pérdidas económicas. Esta misma problemática se puede aplicar a muchos otros ámbitos, tales como, invernaderos, salas de congelación, salas de cría de animales, etc.

2.2 Listado de requisitos y especificaciones.

Una vez planteado el problema es el momento de concretar una lista de los requisitos que tiene cumplir el sistema de control.

Requisitos del sistema a diseñar e implementar:

- **Control remoto de las variables de temperatura y humedad en la cámara de curación:** El sistema tiene que ser capaz de obtener los datos de temperatura y humedad en tiempo real, permitiendo además su modificación a través de la acción de diferentes actuadores.

- **Climatización de la cámara de curación:** El sistema de control tiene que ser capaz de mantener la temperatura en un rango de 12°C – 14°C y que la humedad no supere el 85%.
- **Implementación sin necesidad de obra en la instalación:** La comunicación entre los sensores, actuadores y sistema de control tiene que ser inalámbrica, facilitando así su implementación sin necesidad de hacer obra en la quesería.
- **Ahorro energético:** El controlador tiene que tener en cuenta las condiciones meteorológicas externas y utilizarlas para reducir el consumo energético.
- **Interfaz gráfica:**
 - El manejo del sistema se tiene que realizar de manera visual con PC y Smartphone.
 - Los datos obtenidos se tienen que mostrar gráficamente para facilitar su análisis.
- **Notificar anomalías:** En el caso de superar un rango definido de temperatura o humedad el sistema tiene que enviar una notificación “push” al Smartphone del usuario avisando de la problemática.
- **Software libre:** Se requiere el empleo de software libre para el control y comunicaciones del sistema con el fin de fomentar la cooperación horizontal. Además, se quiere evitar en la mayor medida posible la dependencia de una compañía en particular, abriendo el abanico de opciones y reduciendo los costes del proyecto.
- **Sistema fácilmente escalable:** Como consecuencia de emplear software libre en el control se quiere dar una alternativa en el crecimiento del mundo IoT. Actualmente se están creando decenas de aplicaciones de control remoto sólo compatibles con sus propios dispositivos, uno de los objetivos es evitar esa dependencia y crear un sistema capaz de interactuar con múltiples dispositivos del panorama IoT internacional.

2.3 Descripción del sistema

Una vez establecidas las especificaciones requeridas por el cliente y previo a realizar el análisis del sistema se va a realizar una descripción del sistema completo, con el fin de facilitar la comprensión del documento.

La Figura 2-1 - Diagrama Figura 2-1 muestra el diagrama básico del sistema a diseñar e implementar. Con una rápida visualización se puede observar que el sistema de control recibe los datos de entrada de un sensor de temperatura y humedad, los analiza y manda una orden al actuador con el fin de cumplir con las especificaciones del sistema descritas en el [apartado 2.2](#). Para poder visualizar y acceder remotamente al sistema se ha establecido una comunicación externa a través de internet, en el [apartado 2.4.6](#) se analizan las diferentes opciones.

En el diagrama de la Figura 2-2 se muestra especial atención a la cámara de curación, siendo en nuestro caso la planta del sistema a controlar. Concretamente el control se realiza a través de un actuador capaz de activar y desactivar el motor de frío de la cámara.

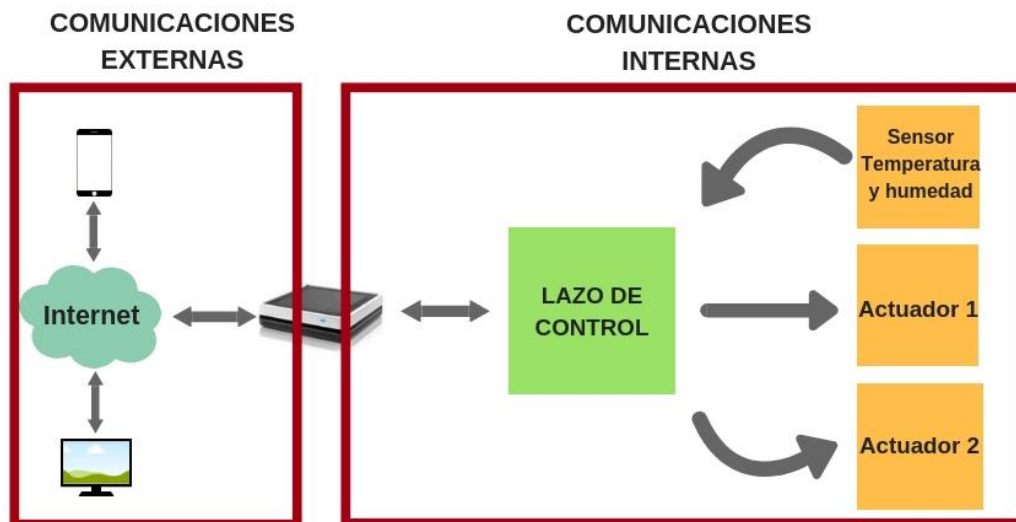


Figura 2-1 - Diagrama del sistema

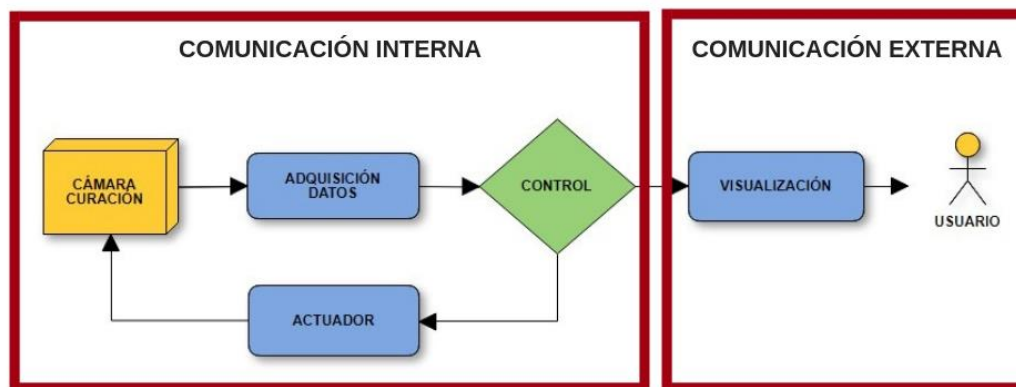


Figura 2-2 - Diagrama básico del sistema

2.4 Análisis del sistema

Previo a realizar el diseño final de nuestro sistema vamos a hacer un análisis detallado de las herramientas de las que se dispone, así como de los elementos a tener en cuenta en el futuro diseño.

Los elementos que requieren un análisis previo son:

- **Cámara de curación**
- **Adquisición de datos**
- **Comunicaciones internas**
- **Sistema de control**
- **Actuadores**
- **Comunicaciones externas**

Todos estos elementos están desglosados a lo largo del capítulo con el fin de documentar las opciones que ofrece la tecnología actual.

2.4.1 Descripción de la cámara de curación

Para comprender correctamente la problemática anteriormente planteada es indispensable conocer las características del local donde se quiere implementar el sistema. Siendo este la cámara de curación de los quesos, que se muestra en la Figura 2-3.



Figura 2-3 - Cámara de curación de quesos

Es en la cámara de curación donde se efectúa el proceso de maduración de los quesos. En el caso estudiado, dependiendo del tipo de queso que se elabore, pueden estar entre 2 y 8 meses hasta su comercialización. La cámara que se va a controlar tiene 50 m² y consta de un equipo de frío dimensionado para dicho tamaño.

Actualmente el control del equipo de frío se realiza mediante un PID del fabricante CAREL, concretamente con dos IR33V9HR20, uno para temperatura y otro para humedad, mostrados en la Figura 2-4.

El sensor de temperatura y humedad empleado en estos momentos es el DPWC110000, mostrada en la Figura 2-5.

La decisión de no emplear estos elementos para el nuevo sistema de control remoto se debe a su incapacidad tecnológica para comunicarse de manera inalámbrica con el sistema, así como disponer de una alternativa en el caso de que se produzcan fallos en el nuevo sistema de control inalámbrico.



Figura 2-4 - PID IR33

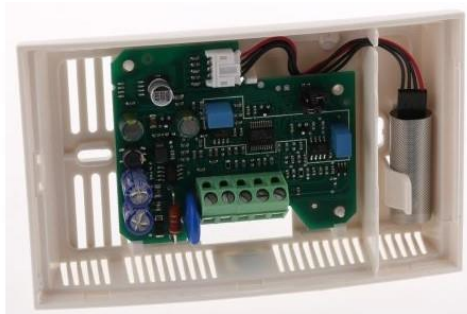


Figura 2-5 - Sensor termo-higrómetro CAREL

Para situar correctamente la superficie de la cámara de curación se han realizado unos planos 3D de las instalaciones (ver Figura 2-6).

2.4.2 Adquisición de datos. Sensores y dispositivos

Como se ha mencionado anteriormente las variables de entrada que se van a controlar son la temperatura y la humedad relativa. Es por tanto interesante analizar las opciones que ofrece el mercado en este tipo de sensores. El análisis se dividirá en dos partes, en primer lugar, analizaremos los sensores básicos y posteriormente dispositivos comerciales con los sensores anteriores integrados.

2.4.2.1 Sensores básicos

Los criterios que se van a utilizar para analizar los sensores de temperatura y humedad son los siguientes:

- **Error de medida:** Posible fallo en el resultado de la medida. Puede variar según el rango de trabajo en el que se encuentre. También conocido como precisión de la medida.
- **Resolución medida:** Número de decimales en la precisión de la medida obtenida.
- **Rango valores:** Capacidad de trabajar en el rango de medidas de interés.
- **Tiempo de respuesta:** Velocidad en la toma de medidas de manera constante.
- **Precio:** Precio de adquisición.

a) Sensor de temperatura “ds18b20 / formato sonda”

Sensor digital de temperatura con protección de acero inoxidable y cableado de hasta 2 metros de longitud. Dichas características le hacen ser el sensor ideal para medir temperatura en líquidos.

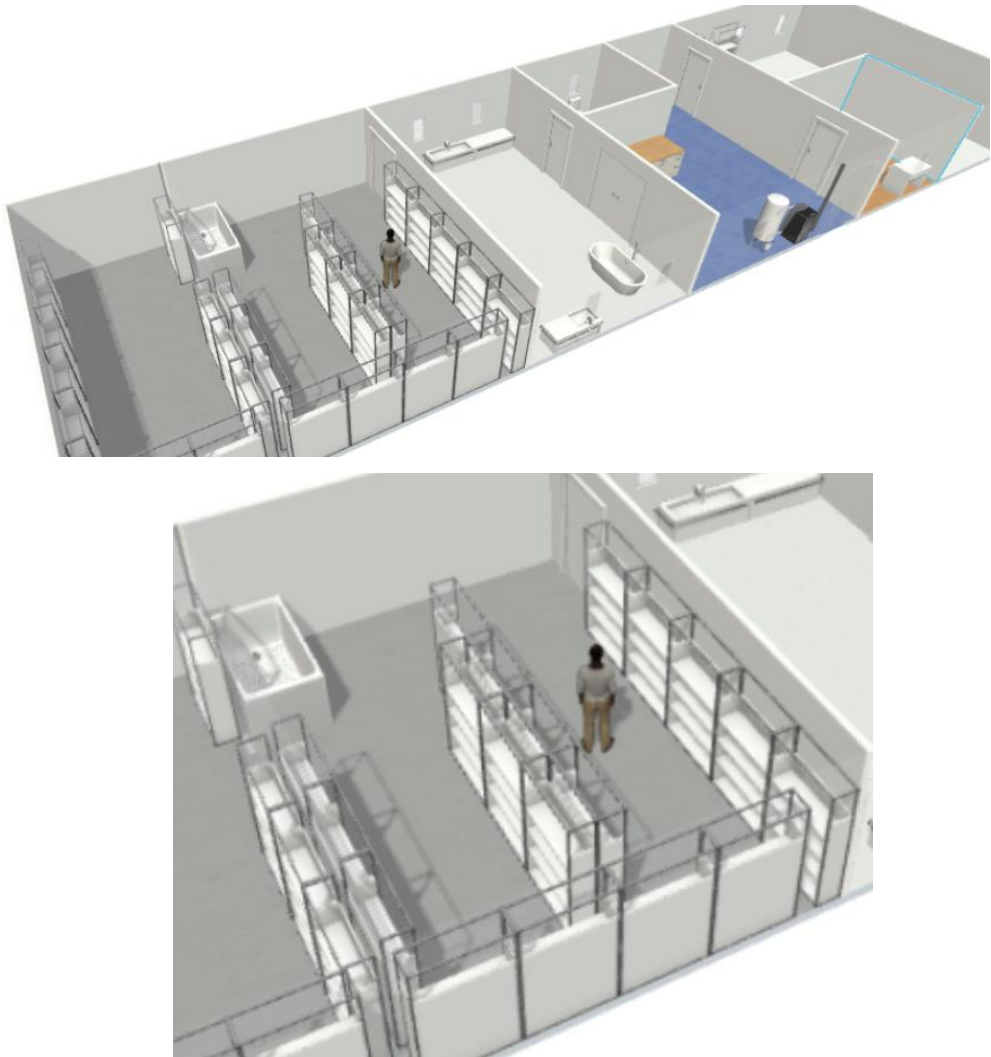


Figura 2-6 - Diseño 3D de las instalaciones

Para la comunicación emplea el protocolo “1-Wire”, el cual identifica de forma única cada sensor, permitiendo así conectar múltiples sensores a un solo pin digital. Para saber más datos ver Figura 2-7.



CRITERIO	DATO
Error (-10°C a 85°C)	± 0,5°C
Error (-55°C a 125°C)	± 2°C
Rango	-55°C a 125°C
Resolución 12-bit	0,0625°C
Tiempo respuesta	2s
Precio	6€

Figura 2-7 - Sensor DS18B20

b) Sensor de temperatura LM35

Sin duda el sensor LM35 es el más utilizado en la electrónica de carácter didáctico a lo largo de los años. Es un sensor analógico con salida lineal, en el que cada grado centígrado equivale a 10mV de tensión a la salida.

Su simpleza, encapsulado y bajo coste lo convierten en el sensor de temperatura perfecto para realizar pruebas en *protoboard* o aplicaciones sencillas (ver Figura 2-8).

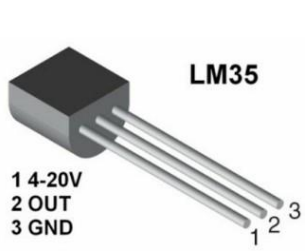
 LM35 1 4-20V 2 OUT 3 GND	CRITERIO	DATO
	Error a 25°C	±0,5°C
	Rango	-55°C a +150°C
	Resolución	-
	Tiempo respuesta	2s
	Precio	1,5€

Figura 2-8 – Sensor LM35

c) Sensor de temperatura y humedad DHT22

La familia de los sensores DHT es de las más utilizadas para proyectos de bajo coste, en los cuales se buscan una buena relación calidad/precio. El sensor DHT22 es una evolución del habitual DHT11, mejorando la precisión y ampliando el rango de funcionamiento.

Es un sensor digital que cuenta con numerosos encapsulados, llegando a incluir la resistencia de *pull-up* para conectar directamente a la entrada del pin digital del microcontrolador. Además, cuenta con numerosas librerías para facilitar la comunicación y evitar preocupaciones respecto al protocolo en cuestión.

En conclusión, es una de las mejores opciones para la toma de datos en proyectos caseros, que no requieran una velocidad rápida de muestreo ni una excesiva precisión (ver Figura 2-9).


 DHT22	CRITERIO	DATO TEMPERATURA	DATO HUMEDAD
	Error (Precisión)	±0,5°C	±2%
	Rango	-40° a 80°C	0 a 100 %
	Resolución	0,1°C	0,1%
	Tiempo respuesta	2s	2s
	Precio	7€	7€

Figura 2-9 - Sensor DHT22

d) Sensor de temperatura y humedad SHT30

El sensor SHT30 es el más preciso de los vistos hasta ahora. El error en la lectura de la temperatura es de solo ±0,3°C (ver Figura 2-10). Por el contrario, es el sensor con mayor tiempo de respuesta, llegando a ser de 8 segundos en cada medida de humedad.

Ofrece una señal de salida digital, siendo su tecnología de comunicación a través del bus I2C. El encapsulado y bajo coste lo hace muy llamativo para proyectos industriales que no requieran un muestreo continuo de datos.



CRITERIO	DATO	DATO
	TEMPERATURA	HUMEDAD
Error (Precisión)	$\pm 0,3^{\circ}\text{C}$	$\pm 2\%$
Rango	-40 a 125°C	0 a 100%
Resolución	12bit	12bit
Tiempo respuesta	2s	8s
Precio	4€	4€

Figura 2-10 - Sensor SHT30

2.4.2.2 Dispositivos comerciales

Una vez comentados los sensores básicos más conocidos en la electrónica de bajo coste, se van a analizar algunos dispositivos comerciales que emplean esos sensores para crear un dispositivo final capaz de medir temperatura, humedad y comunicarse de manera inalámbrica.

En este punto es muy importante aportar datos que permitan tomar la decisión sobre si existe en el mercado algún dispositivo comercial capaz de ofrecer las prestaciones necesarias o por el contrario se debe desarrollar un prototipo para poder alcanzar el objetivo previamente expuesto.

Siguiendo el mismo esquema que en el apartado anterior, se van a establecer unos criterios de análisis que ayudarán a tomar la decisión final en el diseño del sistema:

- **Capacidad de comunicación inalámbrica**
- **Protocolo de comunicación**
- **Consumo de energía**
- **Autonomía**
- **Precio**

a) Sensor AQARA "XIAOMI" de temperatura y humedad

La conocida marca "XIAOMI" se ha introducido en el mundo IoT de la mano de diferentes dispositivos bajo el nombre de AQARA. El sensor de temperatura y humedad que presentan es muy interesante ya que tiene una autonomía de 2 años y es capaz de comunicarse por ZigBee de manera inalámbrica (ver Figura 2-12).

El sensor empleado en este dispositivo es el anteriormente analizado [SHT30](#), el cual obtiene unas prestaciones fiables para el uso que necesitamos. Además la alimentación es aportada por una pila de litio CR2032, fácil de encontrar y sustituir por el usuario.

b) Sensor temperatura y humedad "NEST"

La búsqueda de sensores de temperatura autónomos y con capacidad de comunicación lleva a conocer la compañía americana "NEST" dedicada a la domótica y recientemente comprada por Google. La marca destaca por el elegante diseño de su plataforma y dispositivos (ver Figura 2-13).



CRITERIO	DATO
Capacidad comunicación inalámbrica	SI
Protocolo de comunicación	ZigBee
Consumo de energía	Bajo
Autonomía	2 años
Precio	11€

Figura 2-12 - Sensor "XIAOMI"



CRITERIO	DATO
Capacidad comunicación inalámbrica	SI
Protocolo de comunicación	Bluetooth Low Energy
Consumo de energía	Bajo
Autonomía	2 años
Precio	40€

Figura 2-11 - Sensor "NEST"

Uno de esos dispositivos es un sensor de temperatura y humedad capaz de comunicarse por Bluetooth con el propio termostato de la compañía. Es una opción interesante para usuarios que no quieran complicaciones en la instalación y dispongan de un alto presupuesto.

c) Sensor "SONOFF" TH10/TH16

La última opción analizada proviene de una empresa China, denominada "ITEAD", una empresa que reúne las mejores características, ya que permite obtener sus dispositivos ya configurados para trabajar con su plataforma y a su vez ofrece la posibilidad de modificar su software para adaptarlo a nuestras propias necesidades. Este proceso es muy sencillo, ya que emplean el chip ESP8266 con conectividad WiFi. Incluso el diseño del hardware deja los pines del puerto serie accesibles para su programación.

El dispositivo "SONOFF" TH10/TH16 es un relé de 10A con conectividad WiFi y opción de añadir sensores de temperatura como los analizados anteriormente DHT22 y DS18B20. El gran inconveniente que tiene es que no es autónomo y necesita estar conectado a la red eléctrica (ver Figura 2-13).



CRITERIO	DATO
Capacidad comunicación inalámbrica	SI
Protocolo de comunicación	WiFi 2,4GHz
Consumo de energía	Medio
Autonomía	No disponible
Precio	14€

Figura 2-13 - Sensor "SONOFF"

2.4.3 Comunicaciones internas

Por comunicaciones internas del sistema se entiende en esta memoria todas aquellas comunicaciones que se producen entre el sistema de control, adquisición de datos y actuadores, dejando para más adelante el análisis de las comunicaciones externas entre el usuario y el sistema de control.

Tal y como se ha mencionado en la introducción un requisito importante es establecer una comunicación inalámbrica con el fin de facilitar su implantación sin necesidad de obra. Por esa razón se han descartado muchos de los protocolos más conocidos en el ámbito de la domótica, tales como: KNX o X10.

Para realizar el análisis de las tecnologías de comunicación inalámbricas que ofrece el mercado utilizaremos los siguientes criterios:

- **Consumo de energía**
- **Frecuencia de trabajo**
- **Alcance en campo abierto:** Alcance en metros sin obstáculos de por medio.
- **Velocidad de transmisión**
- **Propiedad:** Libre o privada.

a) GSM/2G/3G/4G



La tecnología móvil está diseñada para operar en largas distancias, contrariamente a las especificaciones de la red de área local necesaria en este proyecto. Además, el consumo de energía es muy alto, lo cual obliga a tener el dispositivo conectado a la red eléctrica en caso de necesitar una larga autonomía.

La principal virtud es su capacidad para enviar gran cantidad de datos con la comunicación 4G (3-10Mbps), así como un alcance de hasta 200Km en la comunicación 3G (ver

Figura 2-14).

CRITERIO	DATO
Consumo de energía	Muy Alto
Frecuencia	Según tecnología
Alcance	Hasta 200 Km
Velocidad transmisión	3-10 Mbps (4G)
Propiedad	Abierta

Figura 2-14 - Tabla de características GSM/2G/3G/4G

b) WiFi



WiFi es el conjunto de especificaciones para redes de área local basadas en el estándar IEEE 802.11. Es el estándar más utilizado para el intercambio de datos de manera inalámbrica en red local, y una opción interesante cuando se necesita una velocidad alta

de transmisión, ya que el nuevo estándar 802.11ac (WiFi 5G) es capaz de alcanzar los 6Gbps en su modo más avanzado (ver Figura 2-15).

El principal inconveniente que tiene es su alto consumo de energía, el cual hace inviable su uso para la comunicación de dispositivos no conectados a la red eléctrica. Sin embargo, puede ser una opción muy útil para actuadores que estén conectados a la red, tales como relés eléctricos, de esa manera se puede permitir un consumo mayor de energía.



CRITERIO	DATO
Consumo de energía	Alto
Frecuencia	2,4 GHz y 5 GHz
Alcance	50m aprox.
Velocidad transmisión	200 Mbps hasta 6Gbps
Propiedad	Abierta

Figura 2-15 - Principales estándares WiFi

c) WiFi Halow 802.11AH



Es un novedoso estándar de comunicación inalámbrica que opera a baja frecuencia, especialmente diseñado para la transmisión de pequeños datos con un alto alcance y un bajo consumo de energía (ver Figura 2-16).

Todavía se encuentra en proceso de implementación y habrá que esperar hasta 2020 para verlo integrado en pequeños consumidores.

Su principal característica es que opera en la banda de 900 MHz. Esa baja frecuencia provoca una mayor cobertura e iguala a Bluetooth en el bajo consumo. Será un protocolo de comunicación muy a tener en cuenta en futuros proyectos del mundo IoT.

CRITERIO	DATO
Consumo de energía	Bajo
Frecuencia	900 MHz
Alcance	50-100m aprox.
Velocidad transmisión	Desconocida
Propiedad	Abierta

Figura 2-16 - Características de WiFi Halow

d) Bluetooth

Bluetooth es la tecnología de comunicaciones de corto alcance más utilizada en el mercado. Destaca por su uso en dispositivos de manos libres, así como para la transferencia de datos entre dispositivos electrónicos sin apenas distancia entre ellos.

Es muy interesante conocer el nuevo "Bluetooth Low Energy" (BLE) ya que se utiliza cada vez más en el mundo IoT. Destaca por un consumo de energía reducido conservando un

alcance similar a Bluetooth. Como hemos podido comprobar anteriormente el sensor de temperatura de la marca "NEST" utiliza esta tecnología para su comunicación.

Sin duda, su gran ventaja es su amplia integración en *smartphones*, lo cual lo hace muy llamativo para proyectos de comunicación entre móviles. Sin embargo, al igual que la mayoría de tecnologías de bajo consume BLE no está diseñado para la transferencia de datos pesados (ver Figura 2-17).

CRITERIO	DATO
Consumo de energía	Bajo
Frecuencia	2,4GHz
Alcance	< 100m
Velocidad transmisión	1Mbps
Propiedad	Abierta

Figura 2-17 - Características de Bluetooth

e) Z-Wave

Z-Wave es un protocolo patentado y diseñado especialmente para usos domóticos y automatización del hogar. La tecnología emplea un transceptor de radio frecuencia de baja potencia, concretamente a 900MHz (ver Figura 2-18).

La diferencia principal con WiFi 2,4GHz es que está plenamente optimizado para el envío y recepción de comandos de bajo peso, reduciendo el consumo de energía y alargando la autonomía de los dispositivos.

Su tecnología de red en malla (MESH) es muy interesante, ya que permite a cada dispositivo enviar y recibir órdenes de control, de esta manera se aumenta notablemente el alcance y calidad de la red.

Es una tecnología cara, ya que para poder comercializar un dispositivo con este protocolo es necesario pertenecer a la "Z-Wave Alliance" para que el producto sea revisado y certificado oficialmente. Además, hasta el año 2016 sus propietarios ("Sigma Designs") no hicieron de dominio público su capa de interoperabilidad, lo cual ha provocado que la competencia y comunidad sea menor elevando los precios de venta de los dispositivos. En este sentido es muy relevante ver como la mayoría de grandes empresas no pertenecen a la "Z-Wave Alliance".

CRITERIO	DATO
Consumo de energía	Bajo
Frecuencia	900 MHz
Alcance	< 30m aprox. En malla
Velocidad transmisión	100 kbps
Propiedad	Abierta desde 2016

Figura 2-18 - Características de Z-Wave

f) ZigBee

En el año 2007 sale a la luz ZigBee con la intención de crear un nuevo estándar para ser utilizado en nuevos productos dedicados a la domótica y automatización del hogar, intentando de este modo evitar tener que hacerlo sobre tecnología patentada. Desde sus inicios la disponibilidad fue gratuita, siendo la primera tecnología *open source* con el objetivo de conquistar el mundo IoT.

Las características de esta tecnología son similares a Z-Wave: bajo consumo de energía, topología en malla (*mesh*) y envío de datos poco pesados. Utiliza la banda ISM de 2,4GHz y es capaz de albergar hasta 255 nodos por cada red ZigBee. Además, es la solución más económica, ya que la radio se puede fabricar con muchos menos circuitos analógicos en comparación con sus competidores.

Después de leer la documentación no es extraño ver como importantes compañías: “IKEA”, “OSRAM”, “XIAOMI”, “NOKIA” o “TOSHIBA” apuestan por ser parte de “ZigBee Alliance” introduciendo la tecnología en muchos de sus novedosos dispositivos.

Haciendo un pequeño análisis de las tendencias de búsqueda en Google durante el último año podemos observar como el protocolo ZigBee supera con creces la popularidad del protocolo Z-Wave. La Figura 2-19 muestra gráficamente el resultado de la búsqueda.

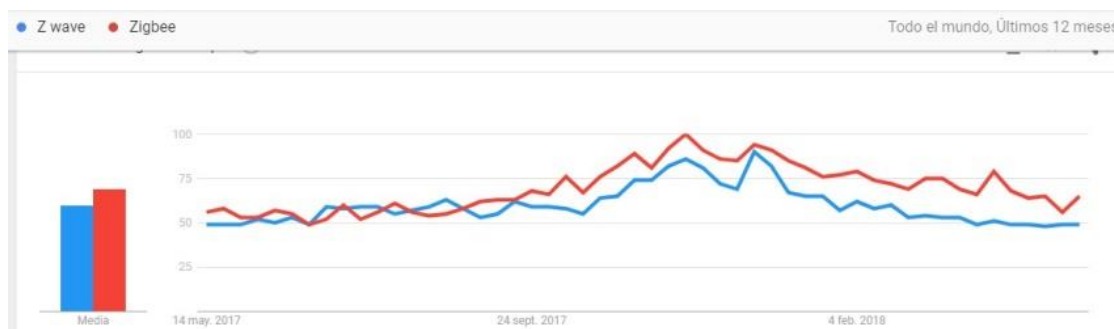


Figura 2-19 - Datos de búsquedas en Google

CRITERIO	DATO
Consumo de energía	Bajo
Frecuencia	2,4 GHz
Alcance	< 100m, más en malla
Velocidad transmisión	250 kbps
Propiedad	Abierta

Figura 2-20 - Características de ZigBee

2.4.4 Sistema de control e interfaz de usuario

El sistema de control es el encargado de gestionar los datos de entrada y a partir de ellos tomar una serie de decisiones de manera autónoma. Esa acción se hará visible en forma de orden enviada al actuador correspondiente. Además tiene que tener la capacidad de albergar un servidor web, ya que queremos disponer de acceso para la operación de nuestro sistema de manera remota.

2.4.4.1 Software

Como requisito descrito en el [apartado 2.2](#), el software de control tiene que ser de propiedad abierta, lo cual permitirá gestionar cualquier dato independientemente de su procedencia. La finalidad es evitar por completo la dependencia de una compañía privada para realizar el proyecto y a la vez tener la opción de emplear dispositivos comerciales que cumplan con nuestras necesidades. De esta manera se amplía el espectro de opciones, mejorando el sistema y reduciendo los costes de implementación.

Las dos principales opciones barajadas son: OpenHAB y “Home Assistant” (HA). Ambas opciones son plataformas de automatización del hogar con capacidad de integración de múltiples dispositivos y totalmente optimizados para ser ejecutados en el hardware de control.

Los criterios que se van a analizar para ayudar a la toma de la decisión final son los siguientes:

- Integración de dispositivos
- Lenguaje de programación
- Comunidad activa
- Interfaz usuario

a) OpenHAB



OpenHAB (“Open Home Automation Bus”) es uno de los softwares libres de control domótico más empleado, creado en Alemania cuenta con una amplia comunidad de desarrolladores, el problema que tiene es que parte de esa comunidad es de habla Alemana, lo cual dificulta la solución de problemas.

Diseñado en Java con una arquitectura modular hace que se puedan integrar múltiples funcionalidades de distintos dispositivos y protocolos. Además con conocimientos de Java se pueden realizar modificaciones de los módulos para adaptarlos a las necesidades del sistema diseñado (ver Figura 2-21).



CRITERIO

DATO

Integración Dispositivos	Alta
Lenguaje de programación	Java
Comunidad activa	Media
Interfaz de Usuario	Anticuada

Figura 2-21 - Interfaz gráfica de OpenHAB

b) “Home Assistant” (HA)

“Home Assistant” es una plataforma de automatización del hogar creada en Python 3, perfectamente optimizada para correr en tarjetas empotradas “Raspberry” y con capacidad infinita de integración de nuevos dispositivos (ver Figura 2-22).

Esa capacidad de integración es su mayor virtud, ya que en caso de no disponer de un componente ya desarrollado se tiene la opción de generar o modificar para adaptarlo a las

necesidades del proyecto en cuestión. Además, es capaz de ejecutar scripts en *Python* diseñados por el usuario, haciendo totalmente personalizable su interfaz gráfica.

Actualmente cuenta con soporte para 1000 dispositivos de diferentes marcas y tecnologías de comunicación, pudiendo acceder a ejemplos de configuración en su amplia web de información:

<https://www.home-assistant.io/components/>

Como la configuración y capacidad de desarrollo es tan grande, su comunidad crece a pasos agigantados. Esa actividad es muy importante, ya que muestra el gran futuro que le espera a esta plataforma en el mundo del control IoT.



CRITERIO

DATO

Integración Dispositivos	Muy alta
Lenguaje de programación	<i>Python 3</i>
Comunidad activa	Muy alta
Interfaz de Usuario	Moderna

Figura 2-22 - Interfaz gráfica de “HomeAssistant”, HA

2.4.4.2 Hardware

Las dos opciones de software que se han barajado están plenamente diseñadas y optimizadas para utilizar una “Raspberry pi” como hardware de ejecución.

Concretamente el modelo “Raspberry pi 3” cumple perfectamente con los requisitos establecidos en este punto:

- **Conectividad WiFi**, estándar 802.11n en la banda de 2,4GHz.
- **Amplia memoria** para albergar el sistema de control y los datos de entrada.
- **Velocidad de procesamiento elevada**
- **Bajo consumo**
- **Bajo costo**
- **Capacidad de albergar software libre**

c) “Raspberry pi 3”

“Raspberry” es un ordenador de placa reducida, tanto en tamaño como en precio. Desarrollado en Reino Unido en 2012 surgió con un objetivo didáctico, aunque actualmente se ha extendido su uso incluso a ámbitos profesionales.

Cuenta con distintas versiones que se han ido actualizando hasta el momento, mejorando enormemente sus prestaciones a lo largo de los años. En esta memoria se van a describir las características del último modelo que ofrece el mercado, denominado “Raspberry pi 3” (ver Figura 2-23). Ha sido un modelo muy perseguido desde el lanzamiento ya que incluye conectividad WiFi y Bluetooth sin necesidad de adquirir un adaptador.



Figura 2-23 – Imagen de la “Raspberry pi 3”

Las principales características técnicas de esta tarjeta son:

- **CPU:** 1,2GHz 64bit quad-core ARMv8.
- **RAM:** 1GB.
- **USB 2.0:** 4 puertos.
- **Salida video:** Conector RCA y HDMI.
- **Salida audio:** Conector de 3,5 mm y HDMI.
- **Almacenamiento:** Tarjeta MicroSD.
- **Conectividad Red:** Ethernet, WiFi 802.11n, Bluetooth 4.1.
- **Periféricos:** 17 GPIO.
- **Consumo:** 800mA.
- **Fuente de alimentación:** 5v vía MicroUSB.

Las tarjetas “Raspberry” cuentan con múltiples opciones en cuanto al software se refiere. El sistema operativo oficial es “Raspbian”, basado en Linux y totalmente optimizado para este tipo de hardware “Raspberry”.

2.4.5 Actuadores

El actuador principal debe permitir activar y desactivar el motor del compresor de la cámara de curación, permitiendo ejercer sobre él un control remoto a través del software de control. Esta función de impedir el paso de corriente se realizará mediante un relé, con el requisito imprescindible de que la señal de control tiene que llegarle de manera inalámbrica.

Un segundo actuador se tiene que encargar del control de una resistencia de calor, ya que en invierno es necesario añadir calor a la instalación.

El requisito de que la comunicación sea inalámbrica condicionará por completo la elección del dispositivo final, ya que se deberá analizar si el mercado ofrece algún dispositivo o por el contrario habría que diseñar un prototipo completo.

Los requisitos del actuador son, por tanto, los siguientes:

- Función ON/OFF (tipo relé)
- Comunicación inalámbrica con nuestro software de control.
- Bajo coste

a) “OSRAM SMART PLUG” ZigBee

La conocida marca francesa de iluminación “OSRAM” se ha adentrado en el mundo IoT con este elegante enchufe inteligente (ver Figura 2-24), capaz de activar y desactivar el paso de energía utilizando el protocolo ZigBee, uno de los protocolos de bajo consumo que hemos analizado en el [apartado 2.4.3](#) de comunicaciones internas.

La principal característica técnica que afecta es la corriente máxima que soporta, siendo de 16 A en este caso.



Figura 2-24 - Actuador “OSRAM SMART PLUG”

b) “SONOFF” SMART SWITCH

Sobre este actuador ya se ha hecho mención en el apartado donde se desgranaban los dispositivos comerciales que ofrece el mercado para medir la temperatura y humedad. La diferencia es que en este caso el dispositivo no lleva ninguna sonda integrada, siendo su única función controlar el relé de 10 A gracias a su conexión WiFi 2,4GHz.



Figura 2-25 - Actuador “SONOFF” de “ITEAD”

c) “TP-LINK” HS100

Es un enchufe de 13 A de corriente máxima controlado por WiFi 2,4GHz.



Figura 2-26 - Actuador “TP-LINK”

2.4.6 Comunicaciones externas

Este último punto de análisis se centrará en la comunicación entre el usuario final y el sistema de control donde poder ver y analizar los datos obtenidos por los sensores.

Como se ha mencionado en el [apartado 2.4.4](#), la plataforma de automatización se instalará en la “Raspberry pi” creando un servidor web local. Por lo tanto, se debe buscar cual es la mejor manera de acceder remotamente a dicho servidor.

En primer lugar, es necesaria la contratación de internet ADSL por parte de cualquier compañía que ofrezca estos servicios. El *router* que proporciona la compañía será el encargado de establecer la comunicación exterior a nuestro sistema de control. En el [capítulo 3.2](#) se detalla el diseño final.

Por defecto, la dirección IP pública del *router* utilizado es dinámica, es decir, cambia de valor según diferentes factores. Por este motivo es imposible acceder de manera remota a la red local si no se pone en práctica una de las dos siguientes opciones:

a) IP fija

Una forma es contratar una dirección IP pública fija. Algunos operadores ofrecen este servicio a cambio de una cuota mensual. De esta manera, se conocerá la IP del *router* y se podrá acceder desde el navegador independientemente del lugar en donde nos encontremos.

b) DNS

La otra forma más común es dar de alta un servicio DNS, lo que hace es traducir la dirección IP pública que tenga el *router* en ese momento por un nombre reconocible para el usuario.

Hay compañías que ofrecen gratuitamente este servicio a pequeña escala, tales como:

- NO-IP
- DynDNS
- Duck DNS

3 Diseño

En este capítulo se detallan punto por punto los elementos elegidos para el diseño completo del sistema de monitorización y control de la quesería artesana, argumentando la razón de su elección y concretando sobre sus características y funciones en el proyecto, a partir de los análisis realizados en el capítulo anterior.

3.1 Red de comunicación interna

En el análisis de comunicaciones internas del [apartado 2.4.3](#) se describieron distintas opciones que ofrece el momento tecnológico actual su implementación.

Finalmente se ha decidido utilizar la tecnología inalámbrica ZigBee, debido a que es (tal y como se analiza en el [capítulo 2.4.3.f](#)) la más adecuada, debido a su bajo consumo, para todas aquellas comunicaciones que requieran larga autonomía en sus dispositivos, como es el caso de los sensores de temperatura y humedad pertenecientes al “Sistema de Adquisición de Datos” (SAD) de la cámara de curación descrita en el [capítulo 2.4.1](#).

Concretamente, las razones de la decisión tomada son las que se listan a continuación:

- Protocolo de bajo consumo con una tasa baja de transferencia de datos.
- Tecnología de código abierto que ha permitido el desarrollo y comercialización de dispositivos de bajo coste debido a que no es necesario el pago de patentes.

Aunque para la mayor parte de las comunicaciones internas se emplea la tecnología ZigBee, no se ha querido limitar el sistema a su uso exclusivo. Como se explica a continuación en el [apartado 3.1.2](#) también se emplea la tecnología WiFi para la comunicación entre el controlador y uno de los actuadores. Hay que destacar que WiFi solo se utiliza para la comunicación de dispositivos conectados a la red eléctrica, ya que su consumo es elevado y no permite una larga autonomía en sus dispositivos.

La Figura 3-1 muestra de manera visual las comunicaciones inalámbricas empleadas para los diferentes sensores y actuadores del sistema.

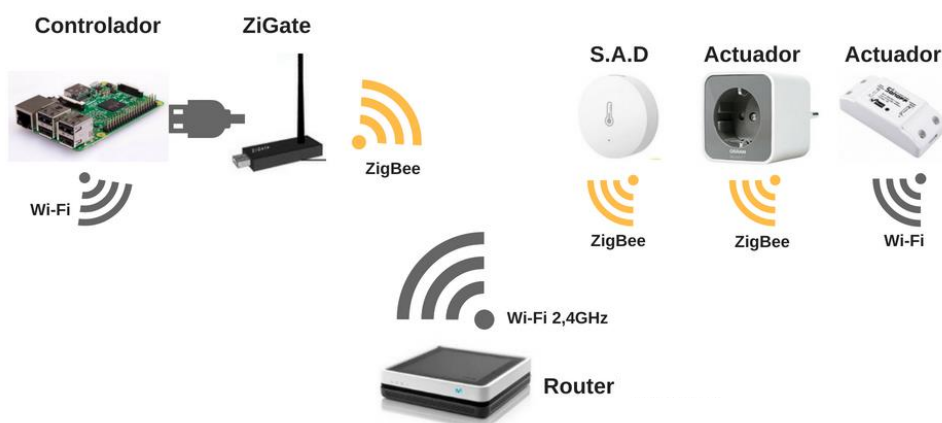


Figura 3-1 - Diagrama de los dispositivos inalámbricos

3.1.1 Zigate

Todas las compañías que emplean la tecnología ZigBee para la comunicación de sus dispositivos del mundo IoT necesitan de un *gateway* (puerta de enlace) para establecer la comunicación entre sus aparatos. Esa necesidad condiciona totalmente el diseño de redes ZigBee, ya que los dispositivos solo funcionan con el *gateway* de su propia marca comercial. Es decir, si se desea controlar dos dispositivos IoT de dos marcas diferentes se necesitaría adquirir *gateways* de los protocolos correspondientes para establecer la comunicación.

Para evitar esta problemática se emplea una puerta de enlace universal ZigBee, denominada Zigate. Se trata de un proyecto con solo 6 meses de vida, diseñado por un desarrollador independiente francés, fabricado y ensamblado en su propio domicilio.

El diseño de Zigate consta de dos partes, un módulo ZigBee MS5168-M05 de “MESHREEM” basado en el microcontrolador ZigBee JN5168 y un módulo USB-TTL.

La Figura 3-2 muestra el interior del dispositivo Zigate con los 2 elementos hardware comentados en el párrafo anterior.



Figura 3-2 - Interior del Zigate

Toda la información y compatibilidad de dispositivos *gateway* se encuentra en la página web oficial del fabricante:

<http://www.zigate.fr>

El uso de una puerta de enlace universal como Zigate añade gran versatilidad al diseño, pues permite elegir entre múltiples dispositivos ZigBee de diferentes compañías para el control del IoT sin los problemas de compatibilidad que hay entre ellas. Este factor permite obtener un diseño más personalizado y una reducción importante de costes.

3.1.2 Comunicación controlador – sensores

En el diagrama de la Figura 3-3 se puede observar el empleo de Zigate como puerta de enlace entre el controlador y los sensores de temperatura y humedad.

La conexión entre el controlador y Zigate se hace a través del puerto USB. Teniendo en cuenta que el hardware de control (“Raspberry pi 3”) dispone de 4 entradas USB, esto no supone ningún problema para su conexión.

Como se ha comentado al inicio del apartado, uno de los factores más importantes en la elección de esta comunicación es lograr un bajo consumo del sensor de temperatura y

humedad, pues al no disponer de alimentación a la red eléctrica se necesita alargar su autonomía lo máximo posible.

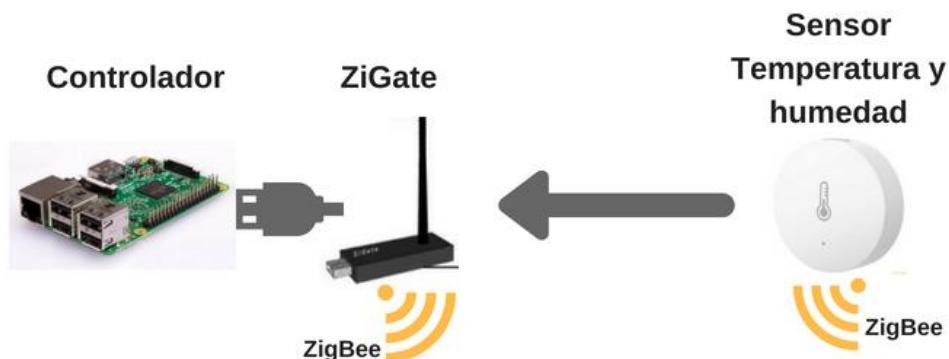


Figura 3-3 - Diagrama de comunicación controlador – sensores

3.1.3 Comunicación controlador – actuador

Una vez evaluados los datos de entrada por el controlador, la decisión de activar o desactivar el motor de frío/calor de la cámara de curación se hace efectiva a través de una orden al actuador correspondiente.

La Figura 3-4 muestra el diagrama de conexionado entre el controlador y los actuadores.

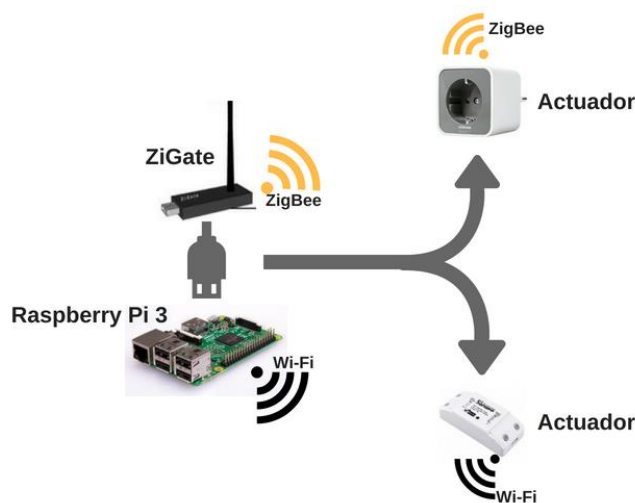


Figura 3-4 - Diagrama de comunicación controlador-actuador

En ese caso los actuadores están conectados a la red eléctrica, por lo que el bajo consumo para alargar la autonomía no es un factor a tener en cuenta en el diseño. Por esa razón, tal y como se comentó al inicio del [apartado 3.1](#) se emplea un actuador con conectividad WiFi. En el [capítulo 3.4](#) se describe del dispositivo elegido en concreto.

Concretamente el actuador mencionado utiliza el protocolo MQTT. MQTT es un protocolo ligero, empleado en la transmisión de mensajes pequeños entre redes de sensores, actuadores, etc.

MQTT está localizado en las capas superiores del modelo OSI (“Open System Interconnection”) y se apoya en la capa de transporte TCP y en la capa de red IP. Debido a ello, la comunicación vía MQTT siempre necesita de una pila TCP/IP para su funcionamiento (ver Figura 3-5).



Figura 3-5 - Capas del protocolo MQTT

Tal y como se describe posteriormente en el apartado de implementación del actuador ([capítulo 4.1.6](#)), este protocolo se basa en el mecanismo de “publicación-suscripción” de mensajes, que se identifican por temas (*topics*). Cualquier dispositivo puede enviar o recibir la información si está suscrito al *topic* (tema) correspondiente. Para llevar a cabo este sistema de comunicación es necesario instalar un *broker* encargado de gestionar el tránsito de mensajes por protocolo MQTT.

3.1.4 Comunicación controlador - *router*

La comunicación entre el controlador y el *router* que conecta esta red interna con la exterior es la que más velocidad de transmisión requiere, ya que todas las consultas desde el exterior hacia el software de control y viceversa tienen que pasar por esta vía. Para optimizar la velocidad de transmisión se ha optado por realizar la conexión entre el *router* y el controlador mediante cable Ethernet.

Es muy importante tener identificada la dirección IP asignada en la red interna al hardware de control, ya que todos los dispositivos que emplean la pila TCP/IP en sus protocolos de comunicación necesitan de esa dirección para establecer la comunicación entre ellas.

Para saber la dirección IP asignada a cualquier dispositivo de la red WiFi se emplea la APP de Android “FING”.

3.2 Red de comunicación externa

Como se ha mencionado en el [apartado 2.4.6](#) la comunicación externa es la encargada de permitir al usuario final controlar y visualizar todas las variables del sistema. Para ello se emplea internet como vía de conexión, concretamente se ha decidido realizar mediante la contratación de un servicio gratuito de DNS, pues la cuota mensual que supone contratar un servicio de IP fija ha hecho descartar esa opción.

La asignación de una dirección DNS a la dirección IP pública soluciona los problemas de acceso al sistema desde el exterior de la red local. Como se explica posteriormente en la descripción de la implementación del sistema ([capítulo 4.1.7](#)) es indispensable configurar y abrir los puertos del *router* para poder acceder remotamente a la interfaz de control.

La implementación y configuración de este diseño permite acceder al sistema de control de la cámara de curación desde cualquier navegador web, ya sea vía Smartphone, Tablet o PC, simplemente introduciendo la dirección DNS del sistema en el navegador web.

3.3 Sensor de temperatura y humedad

Después de ver y analizar las características de los sensores y dispositivos en el [capítulo 2.4.2](#) se ha decidido utilizar el dispositivo comercial de la compañía “XIAOMI” para medir tanto la temperatura como la humedad (entradas del sistema de control de la cámara de curación). Las razones por las que se ha tomado esta decisión son las siguientes:

- Cualquiera de los dos sensores comerciales de temperatura y humedad cumple con los requerimientos del proyecto, ya que no se necesita mucha precisión en las medidas ni un rápido muestreo de los datos, debido a las características físicas de estas señales y su uso en el sistema de control de interés (las magnitudes de temperatura y humedad no van a sufrir cambios bruscos para la curación).
- Los sensores pueden comunicarse de manera inalámbrica empleando la tecnología ZigBee, que debido a su bajo consumo permite alargar la autonomía de sus baterías hasta los 2 años.
- Bajo coste.

Una vez tomada la decisión del dispositivo empleado se procede a diseñar el sistema completo de adquisición de datos (ver descripción del SAD en la Figura 3-36). Los tres principales componentes del SAD son el sensor de temperatura/humedad SHT30, el microcontrolador inalámbrico ZigBee JN5169 y la antena PCB de 2,4 GHz.

El diseño está basado en el microcontrolador “JN5169”, capaz de comunicarse con el sensor “SHT30” por I2C y emitir/recibir los datos inalámbricamente gracias al transceptor ZigBee de 2,4GHz.

El dispositivo final ensamblado queda según se puede observar en la Figura 3-7.

3.3.1 Datos meteorológicos externos

Como se mencionó en la lista de requisitos del sistema del [capítulo 2.2](#) el sistema tiene que ser capaz de actuar en función de los datos meteorológicos externos. Para conocer esos datos se ha decidido emplear la plataforma “darksky”. Esta plataforma online permite obtener distintos datos meteorológicos en tiempo real introduciendo las coordenadas GPS del lugar a sensar. La configuración se explica detalladamente en el [apartado 4.1.4](#) de implementación.

Estos datos se utilizan para producir un ahorro de energía condicionando la activación del actuador, tal y como se explica en el diseño de las automatizaciones ([capítulo 3.7](#)).

3.3.2 Ubicación de los sensores

Una vez seleccionado el sensor “XIAOMI” para obtener los datos de entrada se ha decidido emplear dos de ellos ubicados en distintos lugares de la cámara de maduración:

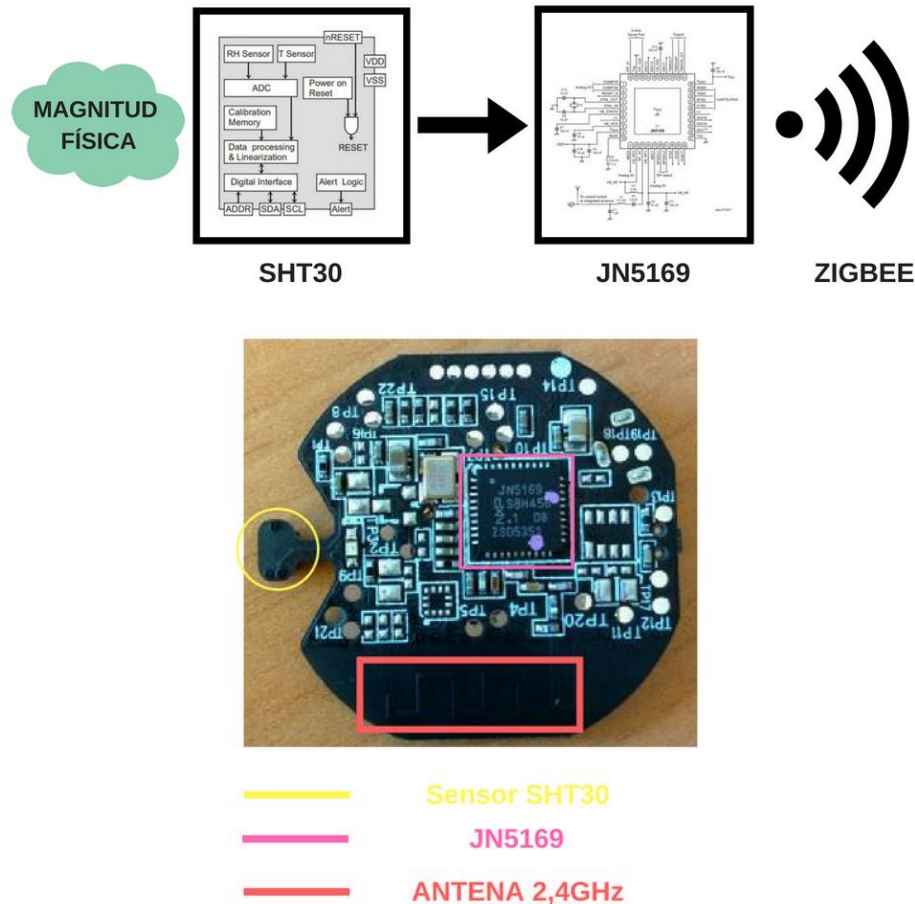


Figura 3-6 - Diagrama de bloques del SAD y vista física interior



Figura 3-7 - Dispositivo final SAD

- El sensor principal está colocado en la parte central de la instalación a una altura de 1,5 metros. Este lugar es el más representativo de las condiciones de temperatura y humedad que sufren los quesos.
- Se ha instalado otro sensor en una de las salidas de frío del equipo de refrigeración, de modo que se pueda detectar cualquier avería en el equipo en el menor tiempo posible.

3.4 Actuadores

El diseño del sistema de control de la cámara de curación que automatiza la quesería incluye dos actuadores diferentes:

- a) **“OSRAM SMART PLUG”**: Se emplea para controlar un radiador eléctrico, de modo que cuando la temperatura exterior sea muy baja se aplique un suplemento de calor a la instalación. La comunicación del controlador con el actuador vía ZigBee permite el control total de la conexión del radiador a la red eléctrica. Las características de este dispositivo se describieron en el [apartado 2.4.5.a](#)
- b) **“SONOFF” SMART SWITCH**: Encargado de activar/desactivar el contactor del motor de frío de la cámara de curación. Su diseño con clemas a la entrada y salida del relé permite una fácil conexión con el cableado del equipo de frío. La comunicación se realiza por WiFi a través del protocolo MQTT, muy utilizado en el mundo IoT. Otro argumento para emplear el actuador “SONOFF” de la marca “ITEAD” es su capacidad de ser modificado por el usuario. Las modificaciones efectuadas en el actuador se detallan en el apartado de implementación del motor de frío ([capítulo 4.1.6](#))

El diagrama de comunicación de los actuadores con el sistema de control de la quesería quedaría como muestra la Figura 3-8.

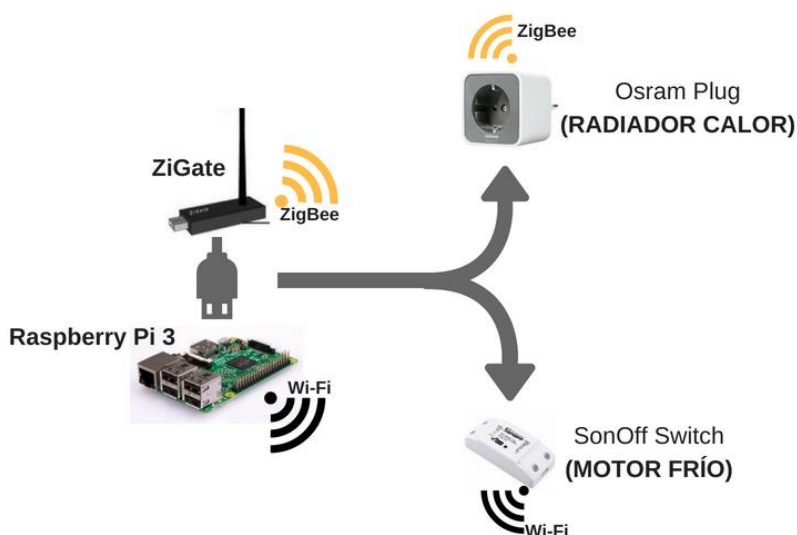


Figura 3-8 - Diagrama de actuadores del sistema

3.5 Hardware

El hardware de control es el encargado de albergar el software *open source* que gestiona el sistema de control de la quesería.

Desde un inicio se ha decidido emplear una “Raspberry pi 3” para este uso, pues tanto sus prestaciones descritas en el [capítulo 2.4.4.2](#), como la versatilidad de su sistema operativo “Raspbian” hacen que sea la herramienta perfecta para albergar el sistema de control.

Dentro de las prestaciones se le da especial utilidad a la conectividad WiFi. En este proyecto, y tal y como se ha explicado en los apartados anteriores, esta conectividad se emplea para la comunicación con el exterior a través de un *router*, así como con alguno de los actuadores (apartados [3.2](#) y [3.4](#)).

Por otro lado, y concretamente para la aplicación de interés, dispone de cuatro ranuras USB permite conectar el *gateway* Zigate sin problemas.

Para alimentar la placa se emplea un transformador (convertor AD-DC) de 5V conectado a la red eléctrica y a otro puerto USB de la tarjeta, quedando ésta como se muestra en la Figura 3-9.



Figura 3-9 – Diagrama de conexiones del hardware de control

Una vez montado el sistema operativo “Raspbian” en la tarjeta SD de la “Raspberry” se está en disposición de instalar el software de control tal y como se explica en el [capítulo 4.1](#).

3.6 Software de control

La plataforma seleccionada para llevar a cabo el control del sistema es “Home Assistant” (HA), tras el análisis incluido en el capítulo [2.4.4.1.b](#) anterior.

Este software se instala en la “Raspberry pi”, dando lugar a una plataforma hardware con las siguientes capacidades:

- Capacidad de obtener y actualizar los datos proporcionados por múltiples componentes IoT de diferentes fabricantes (actualmente cuenta con soporte para más de 1000 componentes). Incluyendo la posibilidad de diseñar el propio en *Python 3*.
- Capacidad para manejar y visualizar todos los datos obtenidos de cualquier dispositivo con un navegador web.
- Capacidad para establecer condiciones de funcionamiento, programar escenas, enviar notificaciones, etc. del sistema de automatización IoT diseñado, en función de los datos de entrada obtenidos.

Si bien las dos plataformas de automatización analizadas en el [capítulo 2.4.4](#) cumplían con los requisitos necesarios para el proyecto, la decisión de emplear HA se debe a su evolución y crecimiento en el sector IoT.

La configuración y programación de la aplicación HA se muestra en el [capítulo 4](#), correspondiente con la fase de implementación en la instalación.

3.7 Automatización

“Home Assistant” utiliza un método de automatización de procesos basado en tres reglas:

- **Trigger:** Describe el evento que da lugar a la regla de automatización. Ejemplo: “Cuando son las 7 de la mañana”
- **Condition:** Establece una regla opcional al evento que da lugar a la automatización. Ejemplo: “Y es un día entre semana”

- **Action:** Indica la acción a ejecutar cuando se cumplen las reglas y sus condicionantes. Ejemplo: “Enciende la radio”

Para cumplir con la lista de requisitos establecidas en [el capítulo 2.2](#) se han diseñado las siguientes automatizaciones:

- **Control de variables de temperatura y humedad:** Para el control básico de temperatura y humedad se emplean dos componentes de climatización. Como se explicará en el [apartado 4.3](#), en la implementación de los automatismos se establecerá una temperatura y humedad objetivo con un rango de funcionamiento. De esta manera fijamos la temperatura y humedad a la que queremos que estén los quesos en la cámara de curación.
- **Ahorro energético:** El funcionamiento del climatizador de temperatura está condicionado a la temperatura exterior de la quesería. De este modo, si anochece y la temperatura exterior es inferior a 10°C se aprovecha la inercia del frío de la noche para bajar la temperatura sin necesidad de activar el equipo de frío. Para llevar a cabo esta acción se realiza una consulta a las 00:00 horas para saber si la temperatura exterior es inferior a 10°C y en caso de necesitar frío la cámara de curación no activa el motor de frío. La implementación de esta automatización viene explicada en el [apartado 4.3](#).
- **Notificación anomalías:** Cuando la temperatura o humedad de la cámara de curación estén fuera de un rango establecido se envía una notificación a través de la aplicación de mensajería “Telegram”.
- **Otras:** Se envía también una notificación cuando se produzca un reinicio del sistema de control HA.

Toda la configuración de los automatismos descritos en este apartado se documentan en el [capítulo 4.3](#).

3.8 Interfaz de usuario

Como se ha mencionado en el [apartado 3.6](#), “Home Assistant” dispone de una interfaz gráfica programable en YAML.

Gracias a la conectividad exterior del sistema de automatización diseñado, esta interfaz es consultable desde cualquier navegador web, de modo que el sistema de control es automático, pero puede monitorizarse en cualquier momento.

Para permitir la tarea de monitorización se ha diseñado una interfaz web con dos pestañas bien diferenciadas:

- **Resumen de datos:** Pestaña donde se pueden visualizar los datos obtenidos de los sensores de temperatura y humedad, configurar la temperatura de la cámara de curación o incluso activar y desactivar manualmente los actuadores en tiempo real. El resultado del diseño se muestra en la Figura 3-10.



Figura 3-10 - Pestaña principal de “Home Assistant”

- **Historial:** Pestaña donde se muestra de manera gráfica todos los datos obtenidos por los sensores. Siendo el resultado del diseño el mostrado en la Figura 3-11.

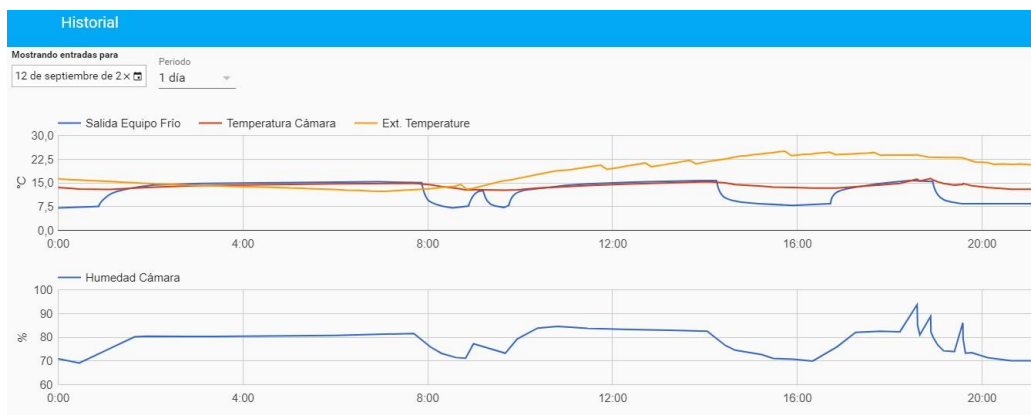


Figura 3-11 - Pestaña del historial de “Home Assistant”

4 Implementación en la quesería artesana

Una vez seleccionados los elementos y comunicaciones que dan forma al sistema se muestra un diagrama global en la Figura 4-1. En esta figura se representa de manera gráfica el dispositivo final elegido, así como el método de comunicación que emplea.

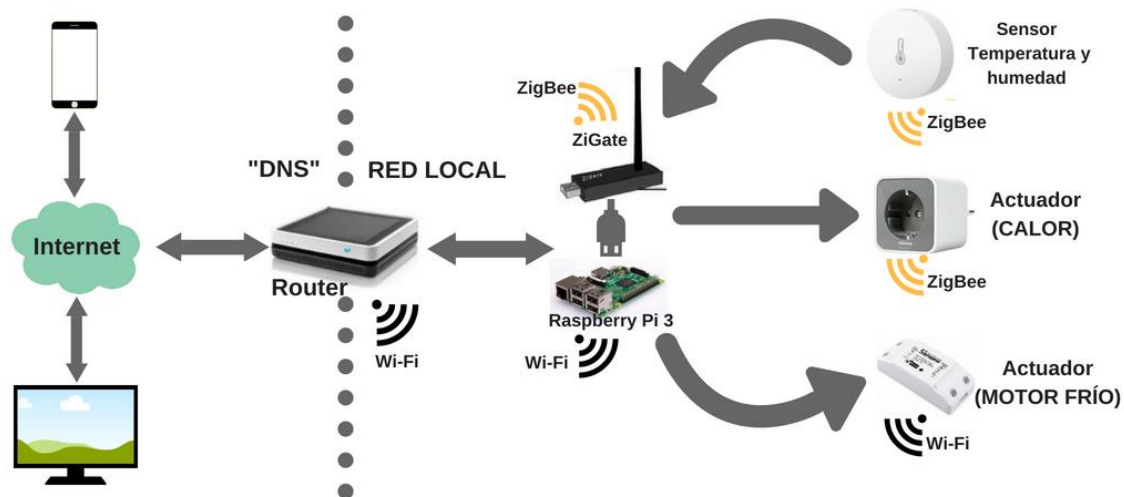


Figura 4-1 - Diagrama global del sistema completo

El capítulo de implementación tiene como finalidad documentar la instalación y configuración del software de control HA. Al finalizar el capítulo se debería tener la información suficiente para reproducir el sistema.

4.1 Configuración de “Home Assistant”

4.1.1 Instalación del sistema operativo

El primer paso en la configuración del sistema de control es la instalación del sistema operativo “Raspbian stretch lite” en la “Raspberry pi 3”. La descarga de este sistema operativo se realiza de manera legal y gratuita en el siguiente enlace:

<https://www.raspberrypi.org/downloads/Raspbian/>

La razón por la que se ha seleccionado “Raspbian stretch lite” y no cualquiera de las otras opciones de “Raspbian” es por la ligereza de la versión, ya que contiene lo mínimo imprescindible instalado por defecto.

“Raspberry” necesita una tarjeta SD de clase 10 (alta velocidad) para instalar el sistema operativo en ella. Una vez descargada la imagen ISO de “Raspbian stretch lite” se procede a montarla en la tarjeta SD.

Para realizar este proceso se ha usado el programa “Etcher”, un proyecto *open source* que permite la grabación de archivos en tarjetas SD de manera segura y gratuita, con página web oficial:

<https://etcher.io/>

Será, por tanto, necesario instalar “Etcher” desde la página web indicada, y ejecutarla para cargar la imagen ISO del sistema operativo “Raspbian” elegido en la SD a incorporar posteriormente en la tarjeta de control, mediante un interfaz como el que se muestra en la Figura 4-2.

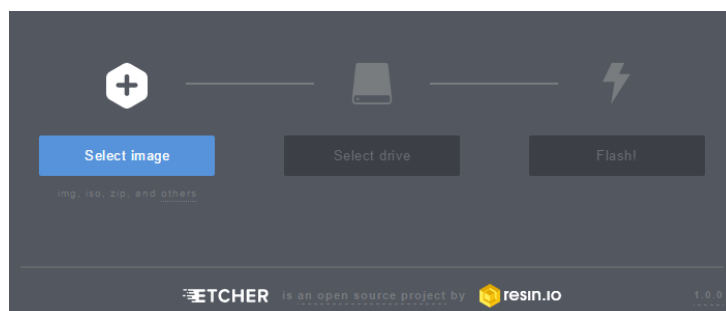


Figura 4-2 - Imagen del programa “Etcher”

Se introduce la tarjeta SD en la ranura correspondiente de la “Raspberry” y al alimentarla se iniciará la instalación del SO de manera automática.

Una vez finalizada la instalación se conecta la “Raspberry” a internet a través de cable Ethernet con el *router* descrito en el [apartado 3.2](#) de comunicaciones externas.

Para navegar por el sistema operativo “Raspbian” se usa el protocolo SSH, un protocolo de administración remota que permite el intercambio de información entre cliente y host a través de internet, por línea de comandos. Con la finalidad de facilitar la navegación por los archivos del sistema operativo desde “Windows” se ha instalado un software con interfaz gráfica, denominado WINSCP, descargable desde su página web oficial:

<https://winscp.net/eng/download.php>

Una vez introducida la IP asignada a la “Raspberry” (ver [apartado 3.1.4](#)) se puede iniciar el intercambio de archivos, mediante la interfaz de WINSCP como la de la Figura 4-3.

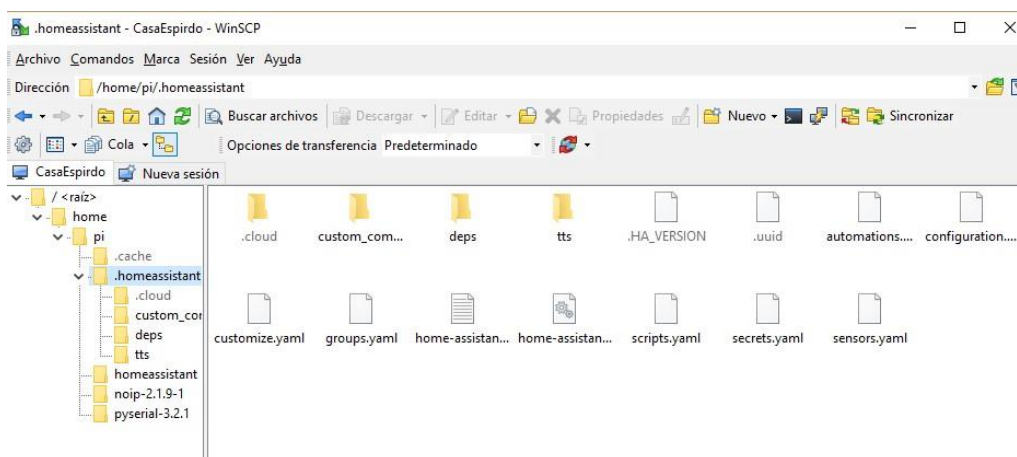


Figura 4-3- Interfaz WINSCP

Es importante tener en cuenta que usuario y contraseña por defecto para acceder a la “Raspberry” por primera vez son: “pi” y “Raspberry” respectivamente. Se recomienda cambiar la contraseña para evitar problemas de seguridad.

Una vez finalizada la configuración de WINSCP, como se ha comentado, se necesita instalar un cliente SSH para poder acceder al terminal y ejecutar los comandos de instalación en “Raspbian”. El cliente SSH instalado en el PC es PUTTY, programa de código abierto y perfectamente optimizado para Windows, con descarga en:

<https://putty.org/>

Para acceder a través de Putty o WinSCP a la “Raspberry” tan solo hay que introducir la dirección IP a la que está conectada en la red local y establecer el puerto 22 (puerto SSH), tal y como muestra la Figura 4-4.

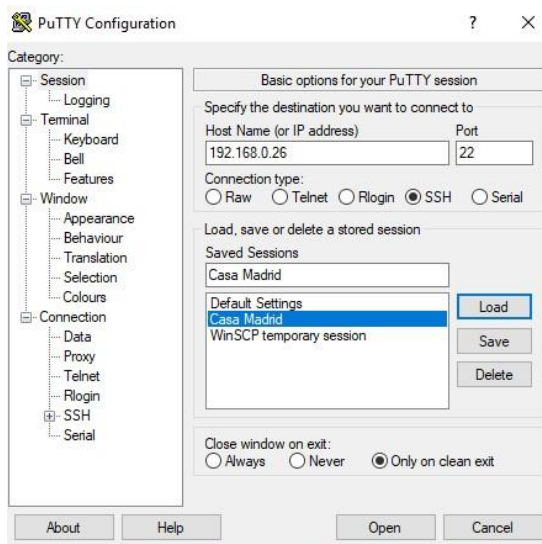


Figura 4-4- Captura del interfaz de Putty

Finalmente, una vez abierto el terminal a través de Putty e introduciendo el usuario y contraseña indicados anteriormente se obtiene por pantalla el resultado mostrado en la Figura 4-5).

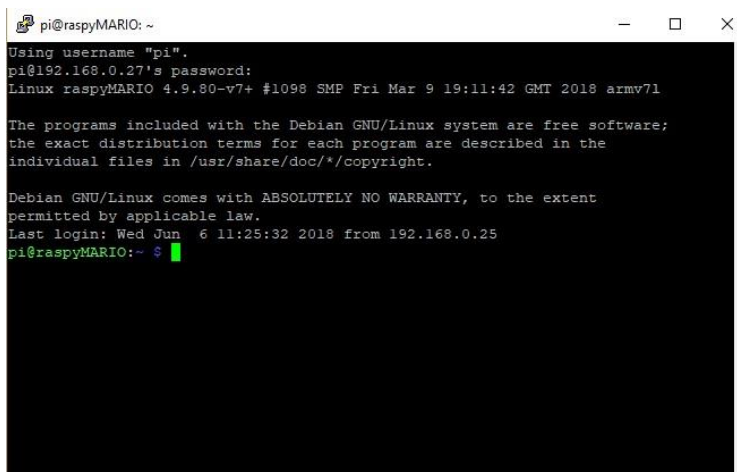


Figura 4-5- Captura de trabajo con terminal con Putty

4.1.2 Instalación de “Home Assistant”

“Home Assistant” cuenta con una amplia guía de instalación y configuración del sistema en su página web oficial:

<https://www.home-assistant.io/>

Para este caso se ha decidido instalar “Home Assistant” en un entorno virtual de “Raspbian”. Se recomienda hacer esta instalación para que el resto de procesos no afecten al correcto funcionamiento de HA.

<https://home-assistant.io/docs/installation/virtualenv/>

Una vez realizada la instalación en el entorno virtual se crean de manera automática los archivos de configuración del sistema. Concretamente, la ubicación del archivo principal de configuración se encuentra en: `/home/pi/.homeassistant`.

Ese archivo de configuración es *configuration.yaml*. Para la edición de este archivo se recomienda utilizar el editor de texto NotePad++, un editor de código fuente libre con soporte para varios lenguajes de programación, con descarga en:

<https://notepad-plus-plus.org/download/v7.5.6.html>.

Con la configuración que viene por defecto ya se puede acceder a la interfaz web de “Home Assistant”. Solo hay que introducir la dirección IP de la “Raspberry” en la red local, seguida del puerto “8123”. Quedando de la siguiente manera:

<http://ip.ad.dre.ss:8123/>

En los siguientes apartados se va a ir editando el archivo de configuración (*configuration.yaml*) para añadir y programar los dispositivos que dan forma al sistema de control.

4.1.3 Configuración del sensor de temperatura y humedad

Como se estableció en el [apartado 3.3](#) del diseño, el sensor de temperatura y humedad seleccionado utiliza la tecnología de comunicación “ZigBee”, por lo tanto todos los datos provenientes de los sensores se reciben a través de la pasarela universal Zigate.

Como Zigate es un proyecto nuevo y con poco recorrido todavía no se ha creado un componente estable en la plataforma “Home Assistant”, por esa razón es necesario añadir el código *Python* del nuevo componente denominado Zigate. Para llevar a cabo esta acción se ha descargado el código creado por un usuario de la plataforma “github”. Descarga en:

https://github.com/elric91/homeassistant_zigate.

Una vez descargados todos los archivos tan solo hay que copiarlos dentro de la carpeta *custom_components*, para posteriormente adaptar el archivo *configuration.yaml*.

- **Componente personalizado:** La Figura 4-6 muestra los archivos del componente Zigate en la carpeta correspondiente *custom_components*.

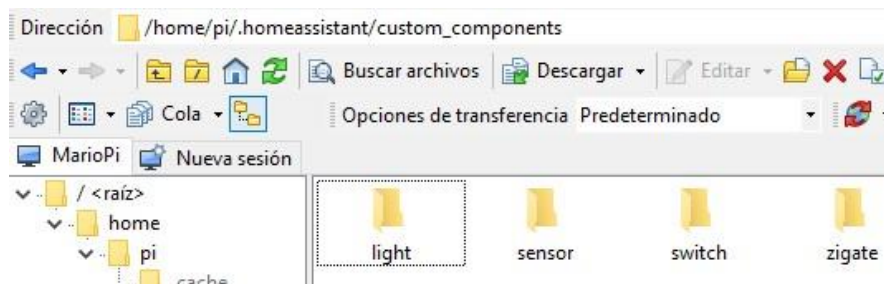


Figura 4-6- Archivos del componente Zigate

En la Figura 4-6 se observa que el componente dispone de tres modelos de empleo, como luz, sensor o interruptor. En este caso se emplea la configuración de Zigate para la recepción de datos de un sensor.

- **Configuration.yaml:** Tal y como se puede observar en la Figura 4-7, la configuración del sensor necesita de una dirección específica proporcionada por el dispositivo. Esa dirección es el código que utiliza Zigate para saber en todo momento de quien procede el dato y establecer la comunicación.

```

34 - platform: zigate
35     name: 'Temperatura Cámara'
36     address: fb8301
37     default_state: temperature
38     default_unit: '°C'
39
40 - platform: zigate
41     name: 'Humedad Cámara'
42     address: fb8301
43     default_state: humidity
44     default_unit: '%'
45

```

Figura 4-7-Configuración de sensores Zigate

La forma más sencilla de averiguar esa dirección es realizar una llamada al comando: *zigate.permit_join* en la ventana de servicios de “Home Assistant” (ver Figura 4-8).

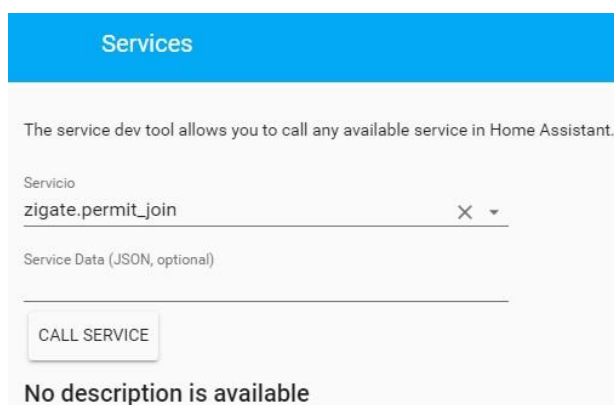


Figura 4-8- Ventana de servicios de HA

De esta manera, durante 30 segundos, se configura Zigate en modo "escucha", mostrando por pantalla el valor de las direcciones de los dispositivos "ZigBee" que han enviado información en ese periodo de tiempo.

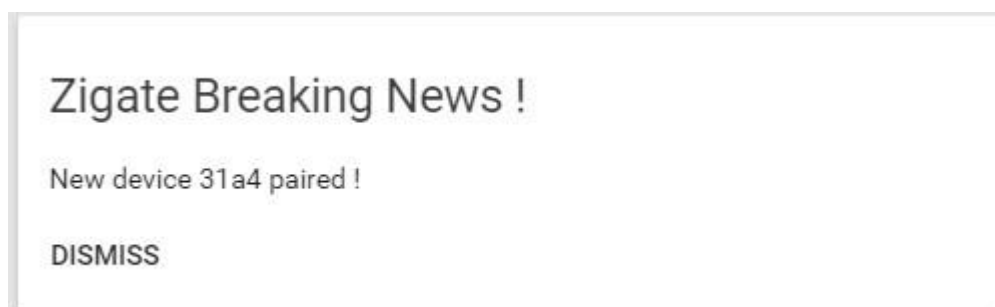


Figura 4-9- Valor de la dirección del dispositivo ZigBee

El resto de la configuración del componente de la Figura 4-7 solo indica el nombre identificativo que aparecerá por pantalla, si el dato es temperatura o humedad y la unidad de medida.

Una vez adaptado el archivo de configuración a los sensores empleados se pueden ver los datos obtenidos a través de la interfaz web de "Home Assistant" (ver Figura 4-10).



Figura 4-10- Visualización del sensor de temperatura y humedad

Más adelante, en el [apartado 4.3](#) se emplean los datos de temperatura y humedad obtenidos para realizar las automatizaciones que darán solución a los requerimientos del sistema.

4.1.4 Configuración del servidor de datos de climatología exterior

Tal y como se mencionó en la lista de requisitos incluida en el [capítulo 2](#), el sistema tiene que hacer partícipe a las condiciones climatológicas exteriores para producir un ahorro energético. Para ello es necesario que "Home Assistant" obtenga estos datos de entrada para posteriormente emplearlos con el fin indicado.

Se ha decidido obtener esos datos a través de la plataforma "darksky", una plataforma que emplea el servidor web www.darksky.net.

"Home Assistant" cuenta con un componente estable y una extensa documentación para configurar correctamente "darksky" en el sistema.

<https://www.home-assistant.io/components/sensor.darksky/>

Como se puede observar en la Figura 4-11, se requiere una “API_key” para disponer de la comunicación continua con la plataforma. Para ello se ha registrado una cuenta particular con un usuario y contraseña que nos permite realizar hasta 1000 consultas de datos diarios de forma gratuita. Una vez obtenida esa llave se configura el archivo *configuration.yaml* según muestra la Figura 4-11.

```

1  #Predicción climatológica exterior a través del servidor "darksky"
2  - platform: darksky
3    api_key: 36158e0e4c8334cc88343d76ccf41bb9
4    name: Ext.
5    update_interval: '00:05:00'
6    units: si
7    monitored_conditions:
8      - summary
9      - hourly_summary
10     - temperature
11     - humidity
12     - apparent_temperature
13     - wind_speed
14     - precip_type
15     - precip_probability

```

Figura 4-11- Archivo de configuración de “darksky”

Una vez reiniciado el sistema se accede a la interfaz web para comprobar la correcta visualización de los datos (ver Figura 4-12).

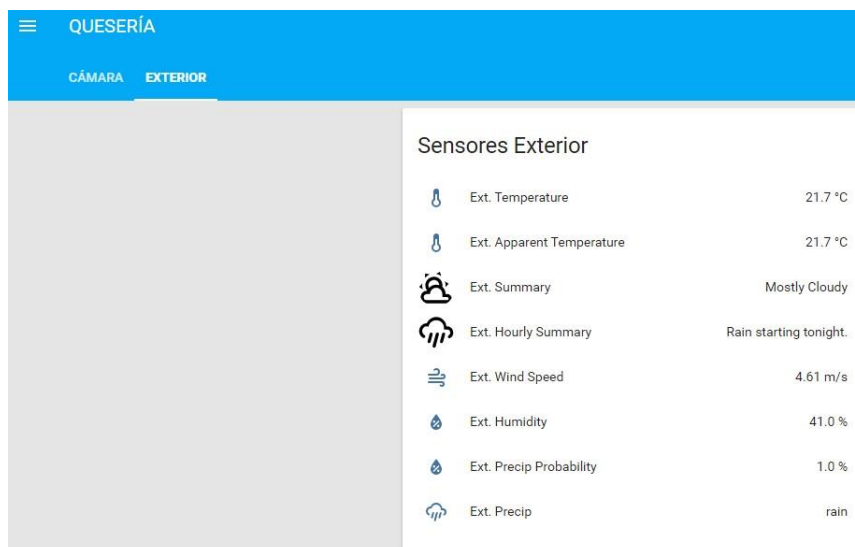


Figura 4-12- Visualización de datos climatológicos

4.1.5 Configuración del actuador radiador de calor

Para el control del radiador se decidió emplear el “SMART PLUG” de “OSRAM” (ver en el capítulo 2), un enchufe con comunicación ZigBee. Al igual que en la configuración de sensores descrita en el [apartado 4.3](#) se emplea la pasarela universal Zigate para establecer la comunicación con nuestro sistema.

En dicha configuración se realizó la instalación del nuevo componente Zigate, por lo tanto, ahora solo es necesario adaptar el archivo *configuration.yaml* para utilizar Zigate como *switch* en lugar de como sensor.

El resultado de esa configuración se muestra en la Figura 4-13.

```

71 switch:
72   - platform: zigate
73     name: 'Resistencia Calor'
74     address: 4e2803
75     default_state: 'state'

```

Figura 4-13- Configuración del interruptor “actuador 1”

Hay que tener en cuenta que la dirección del dispositivo ZigBee es diferente, por lo tanto, hay que realizar un nuevo *permit_join* desde “Home Assistant” para detectar la nueva dirección.

Reiniciando el sistema y accediendo a la interfaz web se visualiza el resultado obtenido, tal y como muestra la Figura 4-14.



Figura 4-14- Visualización del interruptor “resistencia de calor”

4.1.6 Configuración del actuador del motor de frío

El actuador diseñado para el control del equipo de frío no utiliza ZigBee, sino que emplea el protocolo MQTT sobre una pila TCP/IP proporcionada por la tecnología WiFi existente en la instalación.

Como se explicó en el [apartado del diseño de las comunicaciones](#), MQTT necesita de un *broker* para gestionar los mensajes. Se ha decidido emplear el *broker open source* “Mosquitto” instalándolo en la “Raspberry”, de esta manera se mejora la privacidad de los datos, ya que se podría haber utilizado uno en la nube.

- **Configuración del *broker*:** Para instalar “Mosquitto” en el sistema operativo “Raspbian” se ha empleado la documentación obtenida del siguiente enlace web:

<https://swabbster.wordpress.com/2017/06/19/home-assistant-and-mqtt/>.

El documento explica perfectamente como después de instalar el *broker* hay que configurarlo y establecer un usuario y contraseña que nos permita acceder a él de manera segura.

Una vez finalizada la configuración en “Raspbian” se establecen los parámetros que habilitan su funcionamiento en “Home Assistant” (en el archivo *configuration.yaml*), tal y como muestra la Figura 4-15.

```
30 mqtt:
31   broker: 192.168.1.35
32   port: 1883
33   client_id: home-assistant-1
34   username: mosquitto
35   password: mosquitto
```

Figura 4-15- Configuración del *broker* “Mosquitto”

- **Configuración del actuador:** Con la configuración completa del *broker* se puede comenzar a añadir cualquier dispositivo que utilice el protocolo MQTT para su comunicación. En nuestro sistema solo se utiliza este protocolo para la comunicación con el actuador encargado de activar y desactivar el motor de frío de la cámara de curación.

Como se comentó en el apartado [3.1.3](#), MQTT utiliza el método de publicación/suscripción de mensajes a un *topic* (tema) en particular. De esta manera los dispositivos que deseen recibir los datos solo tienen que suscribirse a dicho *topic*.

En el caso del actuador “SONOFF” de “ITEAD” empleado es necesario modificar el software del dispositivo para instalarle un firmware que nos permita configurar cual será el *broker* que coordine sus mensajes y el *topic* al que los tiene que publicar o suscribir. Para llevar a cabo esta acción se ha instalado el firmware de código abierto “Tasmota”.

“Tasmota” es un firmware que te permite controlar y configurar vía web cualquiera de los dispositivos de la marca “ITEAD” basados en el ESP8266. Es un proyecto *open source* con todo el código y documentación disponible en “GitHub”.

<https://github.com/arendst/Sonoff-Tasmota>.

Para programarlo correctamente se ha empleado la documentación del siguiente enlace web:

<http://domology.es/sonoff-tasmota-y-ha/>.

Una vez programado el firmware “Tasmota” en el actuador “SONOFF” Basic disponible, podemos acceder al control de dicho dispositivo, tan solo hay que introducir la dirección IP que le ha sido asignada al actuador, si todo ha ido bien se debería visualizar la pantalla que muestra la Figura 4-16.

Desde el menú principal podemos configurar multitud de parámetros, en nuestro caso se necesita acceder a la configuración del protocolo MQTT, para ello pulsamos en el botón “Configuration” y posteriormente en el botón “Configuration MQTT”. Una vez dentro de la configuración de MQTT se deben indicar los datos del *broker* que gestiona sus mensajes, tales como: dirección IP del *broker*, usuario y contraseña de “Mosquitto” y el *topic* al que publicar o suscribir (ver Figura 4-17).

Con la comunicación por MQTT configurada solo queda añadir el actuador al archivo *configuration.yaml* de “Home Assistant” (ver Figura 4-18).

Reiniciando “Home Assistant” se puede visualizar (ver Figura 4-19) el interruptor de control ya implementado.

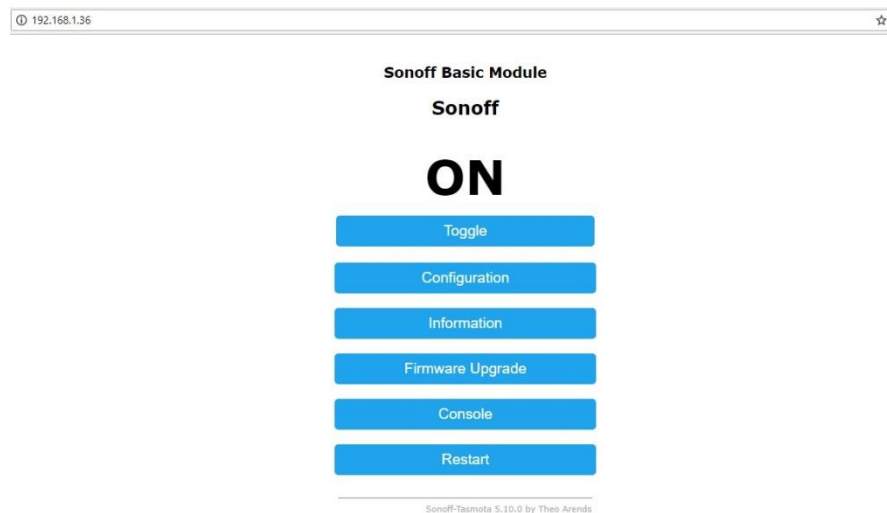


Figura 4-16- Menú principal de “Tasmota”

This image shows the 'MQTT parameters' configuration page. It features several input fields with labels and their current values: 'Host (192.168.0.20)' with value '192.168.1.35', 'Port (1883)' with value '1883', 'Client (DVES_B4D4FA)' with value 'DVES_%06X', 'User (username)' with value 'mosquitto', 'Password' with masked characters '.....', 'Topic = %topic% (sonoff)' with value 'sonoff', and 'Full Topic (%prefix%/ %topic%/)' with value '%prefix%/ %topic%/'. A blue 'Save' button is located at the bottom of the form.

Figura 4-17- Configuración de MQTT en “Tasmota”

4.1.7 Configuración del acceso externo

En el [apartado 3.2](#) se decidió contratar un servicio DNS gratuito a través de la plataforma:

<https://my.noip.com>

Una vez registrado en la plataforma se ha dado de alta un *hostname*, el cual se emplea cuando se quiere acceder al sistema desde el exterior de nuestra red local.


```

6  - platform: mqtt
7    name: "Motor Frio"
8    state_topic: "cmdnd/sonoff/power"
9    command_topic: "cmdnd/sonoff/power"
10   qos: 1
11   payload_on: "ON"
12   payload_off: "OFF"
13   retain: true
14   default_state: 'state'

```

Figura 4-18- Configuración del actuador en HA

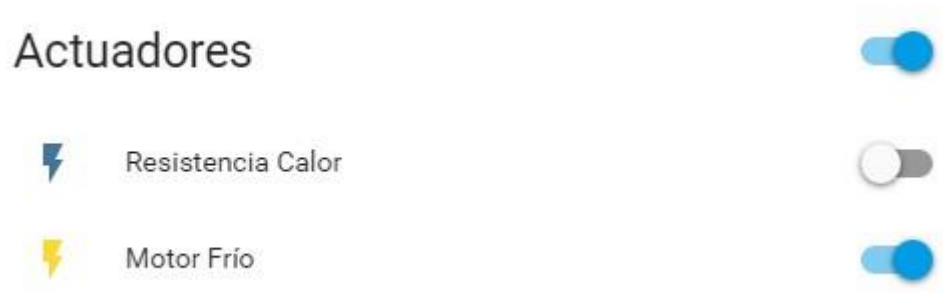


Figura 4-19- Visualización de los interruptores de control de los actuadores

Para poder acceder a la "Raspberry" a través de este *hostname* es necesario instalar el paquete de cliente en "Raspbian". Como apoyo para efectuar correctamente esta instalación se ha utilizado el siguiente artículo:

<http://www.electroensaimada.com/no-ip.html>

Por último es necesario configurar los puertos del *router*, concretamente se abre el puerto "8123", correspondiente al puerto que da accesibilidad a la interfaz gráfica de "Home Assistant" (ver Figura 4-20).

ABRIR PUERTOS

Para que ciertas aplicaciones funcionen, necesitas abrir puertos. Un caso típico es el de los juegos online: P2P, Xbox, PS3... A continuación, te facilitamos un listado de puertos predeterminados, de uso frecuente, para que no tengas que añadirlos de forma manual.

Elige un formato ¿Qué regla de puertos vas a abrir?

☐ Predeterminado
 ☒ Manual
 ☒ Puerto único
☐ Rango de puertos

Dirección IP *	Puertos *	Protocolo	
192.168.1.35	8123	TCP	Añadir
<small>Dirección IP</small>			

Figura 4-20- Interfaz de configuración de los puertos del *router*

4.1.8 Configuración del auto-inicio de HA

Es muy importante configurar “Raspbian” para que “Home Assistant” (HA) se inicie automáticamente al alimentar la “Raspberry”. Para realizar esta configuración se ha empleado la documentación proporcionada por la plataforma oficial de HA:

<https://www.home-assistant.io/docs/autostart/init.d/>

4.2 Notificaciones de “Home Assistant”

Uno de los requisitos del sistema descritos en el [capítulo 2](#) es el envío de notificaciones al teléfono móvil en caso de anomalías en las medidas de temperatura y humedad. Para ello se ha empleado la aplicación de mensajería instantánea “Telegram”.

Una vez instalada esta aplicación en el *smartphone* está en disposición de recibir y enviar mensajes a los contactos que dispongan de ella.

Para llevar a cabo la configuración de este método de notificación se ha empleado la documentación proporcionada por los desarrolladores de “Home Assistant”:

<https://www.home-assistant.io/components/notify.telegram/>

Una vez obtenidos los dos principales parámetros de “Telegram” (Chat_ID, API_TOKEN) se realiza la configuración correspondiente en el archivo *configuration.yaml*, quedando según muestra la Figura 4-21.

```

55 telegram_bot:
56   - platform: polling
57     api_key: !secret Telegram_api_key
58     allowed_chat_ids:
59       - !secret Telegram_chat_id
60     parse_mode: html
61
62 notify:
63   - name: Telegram queseria
64     platform: telegram
65     chat_id: !secret Telegram_chat_id

```

Figura 4-21- Configuración de notificaciones vía “Telegram”

Para comprobar el correcto funcionamiento de las notificaciones de “Telegram” se ha emitido un mensaje a través de “Home Assistant” (ver Figura 4-22).

Recibiendo el mensaje de inmediato en nuestro terminal (ver Figura 4-23). Una vez comprobado el correcto funcionamiento de las notificaciones se puede comenzar con la automatización de los procesos del sistema, tal y como muestra el siguiente punto de la memoria.

4.3 Automatización de “Home Assistant”

En este punto se programan las automatizaciones establecidas en el [capítulo 3.7](#). Al igual que en el resto de la configuración del sistema, el archivo empleado para este caso es *configuration.yaml*. El desglose de las automatizaciones se muestra a continuación, incluyendo

junto a la descripción de la notificación el código incluido en el fichero de configuración mencionado.

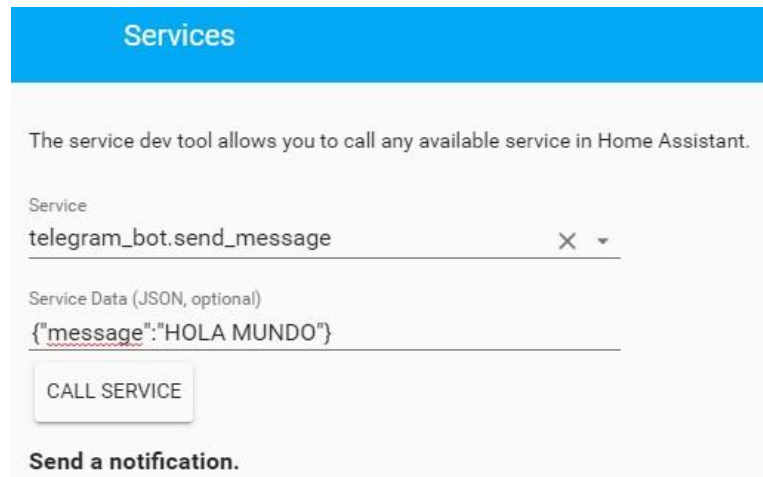


Figura 4-22 - Envío de notificación de prueba en HA

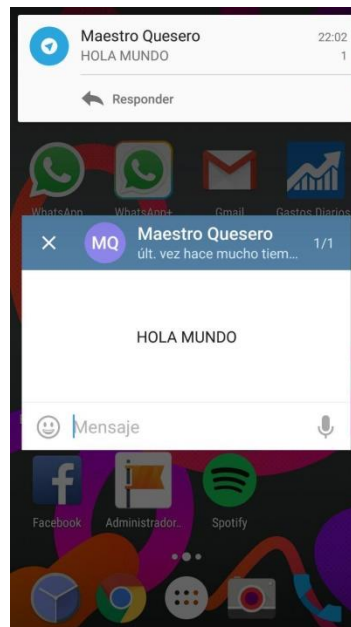


Figura 4-23- Notificación de “Telegram” recibida

4.3.1 Notificación de reinicio de “Home Assistant”

Como se ha comentado en el anterior punto, las notificaciones se llevan a cabo a través de la aplicación de “Telegram”.

Una vez comprobado el correcto envío de mensajes tan solo hay que configurar la automatización que provoca el envío, tal y como muestra la Figura 4-24.

4.3.2 Notificación de anomalías

La automatización efectuada en este apartado se centra en la detección precoz de un posible problema en el equipo de frío.

Por lo tanto, la notificación vía “Telegram” solo se envía en los siguientes casos:

```

2  #-----
3  - alias: 'Reinicio HA'
4  trigger:
5      platform: homeassistant
6      event: start
7  action:
8      service: notify.Telegram_queseria
9      data:
10         message: "Se ha reiniciado Home Assistant"
11  #-----

```

Figura 4-24- Configuración de reinicio de HA

a) Temperatura de la cámara > 15°C (Figura 4-25)

```

15  #-----
16  - alias: 'Temperatura Alta'
17  trigger:
18      platform: numeric_state
19      entity_id: sensor.temperatura_camara
20      above: 15
21  action:
22      service: notify.Telegram_queseria
23      data:
24         message: "Temperatura en cámara superior a 15°C "
25  #-----

```

Figura 4-25- Configuración de notificación de temperatura alta

b) Temperatura de la cámara < 9°C (Figura 4-26)

```

25  #-----
26  - alias: 'Temperatura Baja'
27  trigger:
28      platform: numeric_state
29      entity_id: sensor.temperatura_camara
30      below: 9
31  action:
32      service: notify.Telegram_queseria
33      data:
34         message: "Temperatura en cámara inferior a 9°C "
35  #-----

```

Figura 4-26 – Configuración de notificación de temperatura baja

c) Humedad cámara > 90% (Figura 4-27)

```

35  #-----
36  - alias: 'Humedad Alta'
37  trigger:
38      platform: numeric_state
39      entity_id: sensor.humedad_camara
40      above: 90
41  action:
42      service: notify.Telegram_queseria
43      data:
44         message: "Humedad en cámara superior al 90% "
45  #-----

```

Figura 4-27- Configuración de notificación de humedad alta

d) Humedad de la cámara < 65% (Figura 4-28)

```

45  #-----
46  - alias: 'Humedad Baja'
47  trigger:
48    platform: numeric_state
49    entity_id: sensor.humedad_camara
50    below: 70
51  action:
52    service: notify.Telegram_queseria
53    data:
54      message: "Humedad en cámara inferior al 70% "
55  #-----

```

Figura 4-28- Configuración de notificación de humedad baja**4.3.3 Control de temperatura**

Para controlar la temperatura se ha empleado un componente prediseñado en “Home Assistant” denominado “generic_thermostat”, cuya documentación se puede visualizar en el siguiente link:

https://www.home-assistant.io/components/climate.generic_thermostat/

Tomando como base la documentación se ha adaptado a las necesidades de nuestro sistema, quedando la configuración como muestra la Figura 4-29.

```

67  climate:
68  - platform: generic_thermostat
69    name: "Control Temperatura"
70    ac_mode: true
71    target_sensor: sensor.temperatura_camara
72    heater: switch.motor_frio
73    min_temp: 0
74    max_temp: 30
75    target_temp: 12
76    cold_tolerance: 1
77    min_cycle_duration:
78      seconds: 5

```

Figura 4-29- Configuración del componente *generic_thermostat*

La configuración se ha realizado para establecer 13°C de temperatura por defecto con una histéresis de ±1°C. Se ha seleccionado el sensor que proporciona los datos de temperatura (*sensor.temperatura_camara*) y el actuador (*switch.motor_frio*).

4.3.4 Control de humedad

Para el control de humedad no se dispone de ningún componente implementado por la plataforma “Home Assistant”.

Por lo tanto, se ha optado por realizar una automatización personalizada. Ésta se caracteriza por el establecimiento de unas condiciones de nivel de humedad para activar y desactivar el deshumidificador (ver Figura 4-30).


```

56 - alias: 'Humidificador Encendido'
57   trigger:
58     platform: numeric_state
59     entity_id: sensor.humedad_camara
60     above: 84
61   action:
62     service: switch.turn_on
63     entity_id: switch.motor_frio
64
65 - alias: 'Humidificador Apagado'
66   trigger:
67     platform: numeric_state
68     entity_id: sensor.humedad_camara
69     below: 77
70     for: 00:01:00
71   action:
72     service: switch.turn_off
73     entity_id: switch.motor_frio

```

Figura 4-30- Configuración del control de humedad en HA

4.3.5 Ahorro de energía

El ahorro energético se produce haciendo trabajar lo menos posible al equipo de frío y calor. Para ello se realiza una comparación entre la temperatura exterior y la temperatura en el interior de la cámara de curación.

En caso de que la temperatura exterior sea inferior a 10°C y a su vez el motor de frío esté en funcionamiento se ha automatizado la parada del equipo de frío para aprovechar la inercia térmica de la temperatura exterior (ver la Figura 4-31).

```

75 - alias: 'Ahorro Energía'
76   trigger:
77     platform: numeric_state
78     entity_id: sensor.ext_temperature
79     below: 10
80   condition:
81     condition: and
82     conditions:
83       - condition: state
84         entity_id: switch.motor_frio
85         state: 'on'
86       - condition: time
87         after: '00:00:00'
88         before: '05:00:00'
89   action:
90     - service: switch.turn_off
91       entity_id: switch.motor_frio

```

Figura 4-31- Captura de la automatización para el ahorro energético

4.4 Historial de “Home Assistant”

La configuración del historial es relativamente sencilla, ya que tan solo hay que entrar en nuestro archivo de configuración *configuration.yaml* e ir añadiendo las variables que queramos mostrar gráficamente en la pestaña “Historial” (ver Figura 4-32).

```
44 #Visualiza gráficamente las entidades incluidas
45 history:
46   include:
47     entities:
48       - sensor.temperatura_camara
49       - sensor.humedad_camara
50       - sensor.ext_temperature
51       - sensor.salida_equipo_frio
```

Figura 4-32- Configuración del historial

El resultado es el mostrado en la Figura 4-33.

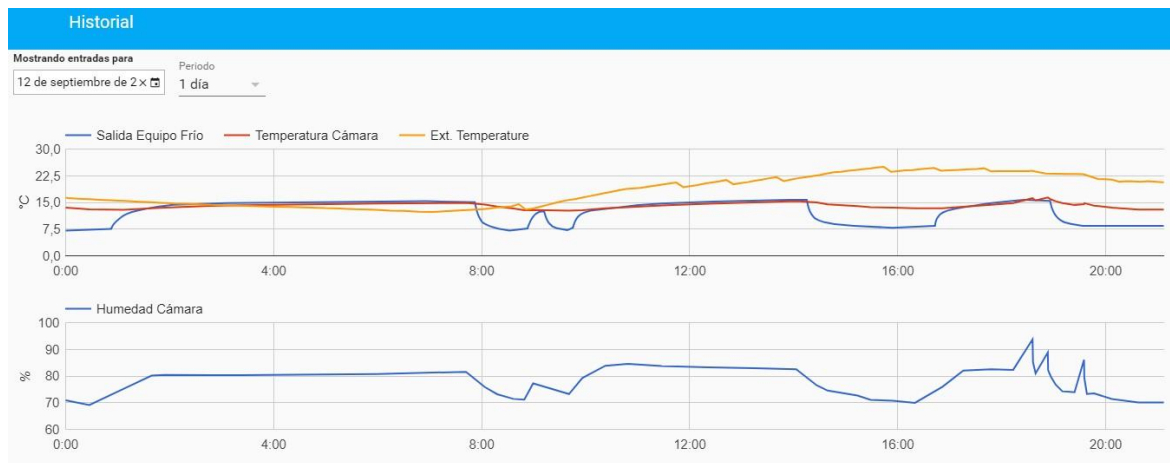


Figura 4-33- Historial gráfico de las medidas

5 Pruebas

Para comprobar el correcto funcionamiento del sistema implantado se han establecido las siguientes dos pruebas que nos permitan evaluar el resultado y las posibles mejoras.

5.1 Temperatura alta en la cámara de curación

Para comprobar la detección de anomalías y su posterior notificación al *smartphone* se ha elevado la temperatura de la cámara por encima de los 15°C (límite establecido para notificar).

La gráfica de la Figura 5-1 muestra el histórico de temperatura donde se puede observar que ha superado los 15°C. En la Figura 5-2 se puede ver la notificación recibida a través de la aplicación de mensajería “Telegram”.



Figura 5-1- Gráfica de temperatura

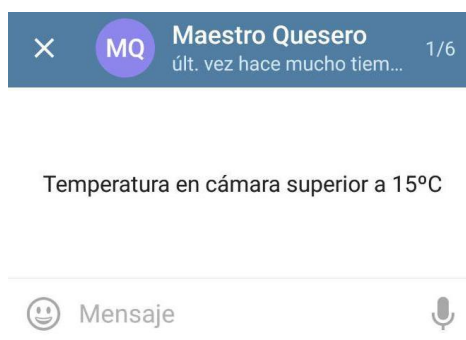


Figura 5-2 – Notificación de “Telegram” de temperatura superior a 15°C

Al realizar esta prueba se ha podido comprobar el correcto cumplimiento de varios de los requisitos establecidos al inicio del documento:

- Interfaz gráfica: Visualización desde Smartphone.

- Interfaz gráfica: Datos en forma gráfica.
- Datos de temperatura y humedad en tiempo real.
- Notificación de anomalías.

5.2 Correcta climatización

Para comprobar la correcta climatización de la instalación se ha realizado una captura de la gráfica resultante de medir la temperatura en el interior de la cámara de curación (ver Figura 5-3). En ella se puede observar como la temperatura permanece en el rango aproximado entre 12°C y 14°C.

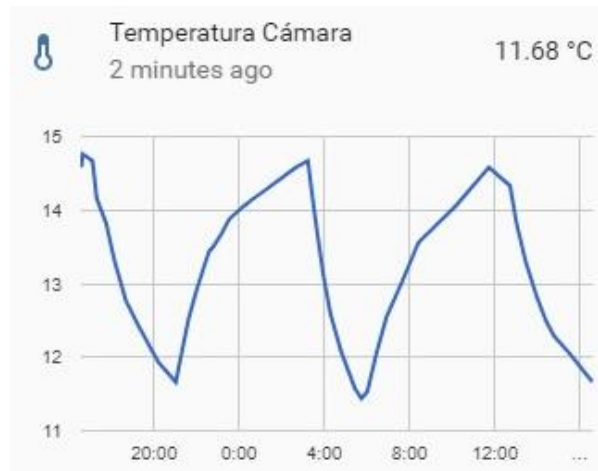


Figura 5-3 - Temperatura en el interior de la cámara de curación

5.3 Correcto funcionamiento del equipo de frío

Mediante el análisis de la temperatura medida por el sensor ubicado en una de las salidas del aire del equipo de frío se puede detectar el correcto funcionamiento de este, ya que la temperatura cuando está encendido tiene que ser de 7° aproximadamente (ver Figura 5-4).



Figura 5-4- Gráfica salida equipo de frío

Este sensor tiene una especial utilidad a la hora de analizar las horas de mejor rendimiento del equipo de frío, ya que puedes comprobar la temperatura mínima que es capaz de alcanzar en función de la temperatura en el exterior de la quesería.

6 Conclusiones y líneas de trabajo abiertas

6.1 Conclusiones

El trabajo de análisis previo a la realización del proyecto ha permitido entender y poner en perspectiva la velocidad a la que evoluciona el mundo IoT. La entrada comercial de grandes empresas en este apartado ha desatado una amplia gama de opciones, eso, unido a las empresas emergentes provenientes de China provoca que los costes de los dispositivos se hayan reducido de manera reseñable. Como se ha demostrado en este proyecto la automatización del hogar ya no es algo elitista económicamente hablando.

Establecer las comunicaciones inalámbricas como prioridad en el desarrollo del proyecto ha permitido analizar y dar cabida a opciones menos comunes como ZigBee. La mayoría de dispositivos hasta hace poco tiempo solo han empleado Z-Wave, Bluetooth o WiFi para sus comunicaciones inalámbricas. El empleo de la tecnología Zigbee ha permitido que las futuras implementaciones sean sencillas de realizar, ya que no será necesario ningún tipo de cableado en la instalación. Cumpliendo de esta manera con uno de los objetivos.

El empleo de la plataforma de software libre “Home Assistant” ha cumplido con creces el objetivo de utilizar código abierto para uso profesional. Es un gran descubrimiento, del cual estaremos hablando mucho en un futuro cercano. Además, asienta el hardware “Raspberry” como la referencia “low cost” de las casas domóticas.

Se ha conseguido poner en valor el movimiento “maker” (“Hazlo tú mismo”) con el empleo de Zigate como puerta de enlace ZigBee, así como con la utilización del software libre “Tasmota” en la configuración del actuador de la compañía “SONOFF”.

Por último, el resultado obtenido en la visualización y control de las variables de temperatura y humedad en la cámara de curación ha sido ampliamente satisfactorio, siendo capaz de detectar anomalías y posibles mejoras en la instalación desde el primer momento. Concretamente, el sistema ha dejado ver la importancia de invertir en mejorar el aislamiento en la cámara, así como la capacidad de detectar las anomalías que permiten realizar un mantenimiento preventivo adecuado del equipo de frío.

6.2 Líneas futuras

Una vez implementado el sistema en la quesería se puede comprobar que este sistema de control es solo el inicio de un proyecto con muchas más opciones. Durante la realización del mismo han ido surgiendo varias ideas que serían viables implantarlas en un futuro cercano.

- **Envío de notificación al bajar el nivel del tanque de Gasoil.** Mediante un sensor de nivel de líquido (tipo boya) y un ESP8266 (microcontrolador con conectividad WiFi) se enviaría una notificación a nuestra interfaz de usuario. De esta manera se podría evitar el despiste humano si no se mira periódicamente el nivel del tanque de gasoil.
- **Control de temperatura de la sala de elaboración.** Implantación del mismo sistema en la sala de elaboración. Se emplearían los mismos sensores de temperatura, solo modificando el actuador correspondiente adaptándolo al nuevo equipo de climatización.

- **Encendido de la caldera de manera remota.** Empleando un interruptor con conectividad WiFi o Zigbee se encendería y apagaría la caldera encargada de calentar la “cuba de cuajar”. Esto permite optimizar el tiempo previo a la llegada de la leche, ya que se podría encender antes de desplazarse a la instalación.
- **Aumentar el ahorro de energía:** Diseñar una automatización que aproveche las horas más baratas de consumo eléctrico.

7 Presupuesto

En este apartado se realiza una estimación del coste total que supone la ejecución del proyecto. En los siguientes puntos aparecen los gastos agrupados según su origen, y en el último se detalla el presupuesto total.

7.1 Material que compone el sistema

CONCEPTO	PRECIO UNIT.	CANTIDAD	SUBTOTAL
"Raspberry pi 3" + Tarjeta SD + Alimentación + Carcasa RP3	54€	1	54 €
Sensor temperatura y humedad "XIAOMI"	11€	2	22 €
Actuador "SONOFF"	5€	1	5 €
Actuador "OSRAM SMART PLUG"	30€	1	30 €
Zigate USB-TTL	49€	1	49 €
TOTAL			160 €

7.2 Recursos hardware

CONCEPTO	PRECIO UNIT.	CANTIDAD	SUBTOTAL
Ordenador portátil Intel Core, i5	550€	1	550 €
FT232 USB to UART	12 €	0	12 €
Material oficina	100 €	1	100 €
TOTAL			662 €

7.3 Recursos software

El empleo de software libre ha hecho que el coste para estos recursos sea muy reducido. Incluso se podría llegar a reducir al 100% si se emplea "OpenOffice" en lugar de "Microsoft Word".

CONCEPTO	PRECIO UNIT.	CANTIDAD	SUBTOTAL
Microsoft Word	60€	1	60€
Software libre	0€	0	0€
TOTAL			60€

7.4 Coste de la mano de obra

La realización de este proyecto ha sido llevada a cabo por las siguientes personas (Home Assistant, 2018):

CONCEPTO	PRECIO UNIT.	CANTIDAD	SUBTOTAL
Ingeniero	50 €	300 horas	15.000€

7.5 Presupuesto de ejecución material

La suma total de los importes del coste de materiales y de la mano de obra, especificados anteriormente, se desglosa a continuación:

CONCEPTO	PRECIO
Material del proyecto	160 €
Recursos hardware	562 €
Recursos software	60 €
Mano de obra	15.000 €
TOTAL	15.782 €

7.6 Importe de la ejecución por contrata

A continuación, se incluyen los gastos derivados del uso de las instalaciones donde se ha llevado a cabo el proyecto, cargas fiscales, gastos financieros, tasas administrativas y derivados de las obligaciones de control del proyecto. De igual forma se incluye el beneficio industrial.

Para cubrir estos gastos se establece un recargo del 22% sobre el importe del presupuesto de ejecución material.

CONCEPTO	PRECIO
22% Coste total de ejecución material	3.472.04 €

7.7 Honorarios facultativos

Se ha fijado en este proyecto un porcentaje del 7% sobre el coste total de ejecución por contrata.

CONCEPTO	PRECIO
7% coste ejecución material	1.104,74 €

7.8 Presupuesto total

CONCEPTO	SUBTOTAL
Presupuesto de ejecución material	15.782 €
Importe ejecución por contrata	3.472,04 €
Honorarios facultativos	1.104,74 €
TOTAL (sin IVA)	20.358,78 €
IVA (21%)	4.275,34 €
TOTAL	24.634,12 €

El presupuesto total del proyecto asciende a la cantidad de veinticuatro mil seiscientos treinta y cuatro euros con doce céntimos.

Alcalá de Henares a Septiembre de 2018.

Firmado: Jorge Triviño Romano

Graduado en Ingeniería Electrónica de Comunicaciones

Referencias

- [1] J. León Almazán, Diseño de módulos software para un sistema avanzado robótico de asistencia a la movilidad basado en entorno de desarrollo robótico ROS, UAH, 2017.
- [2] "glowlabs.co," [Online]. Available: <http://glowlabs.co/wireless-protocols/> [Accessed 2018].
- [3] "Python," [Online]. Available: <https://www.python.org/> [Accessed 2018].
- [4] "Home Assistant" [Online]. Available: <https://www.home-assistant.io> [Accessed 2018].
- [5] "Zigate" [Online]. Available: <https://zigate.fr/> [Accessed 2018].
- [6] "Domoticz" [Online]. Available: <http://www.domoticz.com/> [Accessed 2018].
- [7] "OpenHab" [Online]. Available: <https://www.openhab.org/> [Accessed 2018].
- [8] "Tasmota" [Online]. Available: <https://github.com/arendst/Sonoff-Tasmota/> [Accessed 2018].
- [9] "Blog Ricveal" [Online]. Available: <https://ricveal.com/> [Accessed 2018].
- [10] "SONOFF" [Online]. Available: <http://sonoff.itead.cc/en/> [Accessed 2018].
- [11] R. Vega Alonso, Instalación domótica basada en OPENHAB y Raspberry Pi, Universidad de Valladolid, 2016.
- [12] "Raspbian" [Online]. Available: <https://www.raspberrypi.org/downloads/raspbian/> [Accessed 2018].
- [13] E. González Daza, Red de sensores – Internet de las cosas, Universidad Politécnica de Valencia, 2015.
- [14] "Blog programar fácil, Luis del valle" [Online]. Available: <https://programarfácil.com/esp8266/domotica-sonoff-wifi-espurna/> [Accessed 2018].
- [15] "Domology" [Online]. Available: <http://domology.es/sonoff-tasmota-y-ha/>. [Accessed 2018].
- [16] "Putty" [Online]. Available: <https://putty.org/> [Accessed 2018].
- [17] "RS-online" [Online]. Available: <https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about> [Accessed 2018].
- [18] "Etcher" [Online]. Available: <https://etcher.io/> [Accessed 2018].
- [19] "WinSCP" [Online]. Available: <https://winscp.net/eng/download.php> [Accessed 2018].
- [20] "Noip" [Online]. Available: <https://my.noip.com> [Accessed 2018].
- [21] "Zigbee" [Online]. Available: <https://www.zigbee.org/> [Accessed 2018].

Anexo A. Manual de usuario

El manual de usuario explica brevemente la forma de acceder y navegar en el interfaz de usuario del sistema de control implementado en la quesería.

A.1. Acceso a la interfaz

A.1.1. Desde dentro de la red local

Para acceder desde dentro de nuestra red local es necesario introducir la dirección IP asignada a la “Raspberry” en cualquiera de las aplicaciones de navegación web instalada. Además, hay que añadir el puerto “8123” que emplea la plataforma “Home Assistant”.

direccionip:8123

A.1.2. Desde el exterior de la red local

En este caso hay que introducir la dirección DNS configurada junto con el puerto “8123”.

direcciondns:8123

A.2. Contraseña de acceso

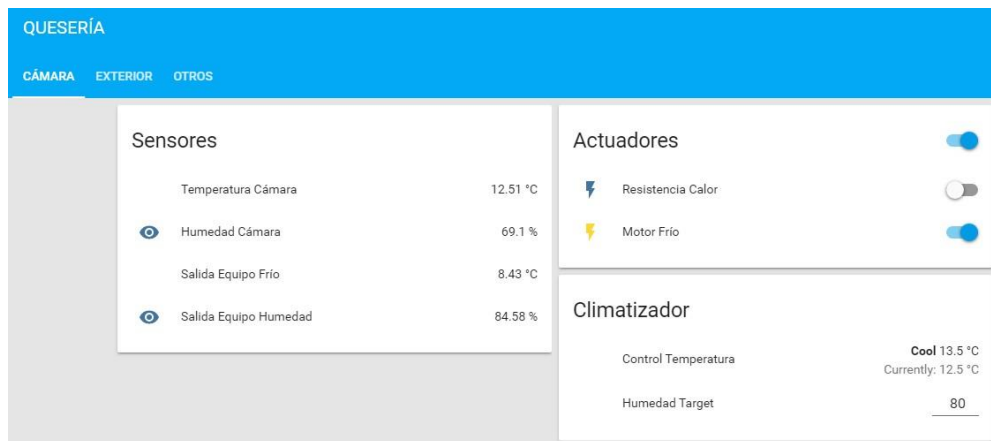
Independientemente de si el acceso se ha realizado desde el exterior o interior de la red local se pedirá una contraseña de acceso a la plataforma web (ver Figura A-1).



Figura A-1- Captura introducir contraseña

A.3. Pantalla de resumen

Una vez validada la contraseña se abre automáticamente la pantalla principal de la interfaz de usuario, según muestra la Figura A-2 .

**Figura A-2- Pantalla resumen**

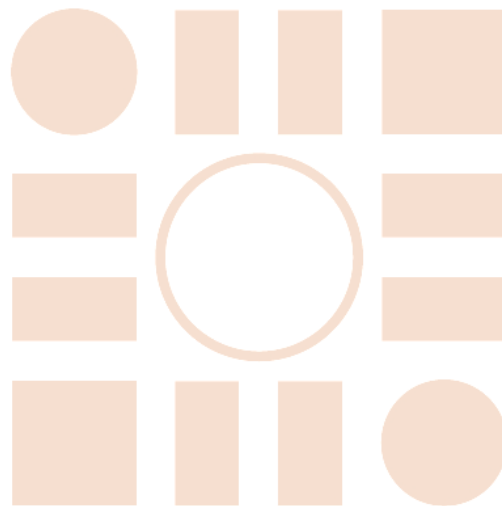
En esta pantalla se pueden configurar manualmente los valores de climatización, además se muestran los datos obtenidos por los sensores en tiempo real.

A.4. Pantalla de historial

La interfaz de usuario cuenta con un apartado dedicado a mostrar de manera gráfica los datos obtenidos por los sensores. En la pestaña superior izquierda se puede seleccionar el día del que se desea ver los datos (ver Figura A-3).

**Figura A-3- Captura del historial**

Universidad de Alcalá
Escuela Politécnica Superior



ESCUELA POLITECNICA
SUPERIOR



Universidad
de Alcalá